正本

經濟部智慧財產局　函

受文者：蒙恬科技股份有限公司（代理人：林火泉　先生）

速別：速件
密等及解密條件：
發文日期：中華民國九十二年九月十六日
發文字號：〈九二〉智專一（一）14005字第〇九二二〇九三三五八〇號
附件：申請證明文件一份

主旨：檢送第〇九二一〇七七七九號專利申請案申請證明文件一份，請　查照。

說明：依九十二年八月二十九日申請書辦理。

正本：蒙恬科技股份有限公司（代理人：林火泉　先生）

副本：

106 臺北市大安區忠孝東路四段三一一號十二樓之一

林火泉　先生

掛號

發文文號：09220933580

局長 蔡練生

依照分層負責規定授權單位主管決行

機關地址：台北市辛亥路二段一八五號三樓
傳　真：（〇二）二七三五二八〇〇
如有疑問請電洽（〇二）二七三八〇〇〇七分機九〇二二

92.9.18 收文章

09220933580

裝　訂　線

c:\A9200312.112

中華民國經濟部智慧財產局

INTELLECTUAL PROPERTY OFFICE
MINISTRY OF ECONOMIC AFFAIRS
REPUBLIC OF CHINA

茲證明所附文件，係本局存檔中原申請案的副本，正確無訛，
其申請資料如下 ：

This is to certify that annexed is a true copy from the records of this office of the application as originally filed which is identified hereunder :

申　請　日 ： 西元 2003 年 04 月 04 日
Application Date

申 請 案 號 ： 092107779
Application No.

申　請　人 ： 蒙恬科技股份有限公司
Applicant(s)

局　　長
Director General

蔡練生

發文日期： 西元 2003 年 9 月 16 日
Issue Date

發文字號：
Serial No. 09220933580

發明專利說明書

（填寫本書件時請先行詳閱申請書後之申請須知，作※記號部分請勿填寫）

※ 申請案號：＿＿＿＿＿＿＿＿ ※IPC分類：＿＿＿＿＿＿＿＿

※ 申請日期：＿＿＿＿＿＿＿＿

壹、發明名稱

（中文）應用於語音辨認之語音模型訓練方法

（英文）＿＿＿＿＿＿＿＿

貳、發明人（共 1 人）

發明人 1 （如發明人超過一人，請填**說明書發明人續頁**）

姓名：（中文） 洪維廷

（英文）＿＿＿＿＿＿＿＿

住居所地址：（中文）新竹市光復路二段2巷47號7樓

（英文）＿＿＿＿＿＿＿＿

國籍：（中文） 中華民國 （英文）＿＿＿＿＿＿＿＿

參、申請人（共 1 人）

申請人 1 （如發明人超過一人，請填**說明書申請人續頁**）

姓名或名稱：（中文）蒙恬科技股份有限公司

（英文）＿＿＿＿＿＿＿＿

住居所或營業所地址：（中文）新竹市光復路二段2巷47號7樓

（英文）＿＿＿＿＿＿＿＿

國籍：（中文）中華民國 （英文）＿＿＿＿＿＿＿＿

代表人：（中文） 蔡義泰

（英文）＿＿＿＿＿＿＿＿

肆、中文發明摘要

本發明提供一種應用於語音辨認之語音模型訓練方法，首先係將輸入語音分離模型化成為一乾淨聲音之密實語音模型及一環境因素模型，而後根據該環境因素模型將輸入語音中之環境雜訊濾除以得到一環境效應抑制的語音訊號，接著再將此語音訊號與密實語音模型，利用鑑別式訓練法演算而得到一高鑑別度且密實的語音訓練模型，以提供語音辨認裝置進行後續之語音辨認處理。因此利用本發明演算得之語音訓練模型同時兼具有強健能力及鑑別能力，進而具有高辨識率之優點，且適用於雜訊環境的補償辨認，並可達到精準之環境效應調適。

伍、英文發明摘要

陸、(一)、本案指定代表圖爲：第__一__圖

(二)、本代表圖之元件代表符號簡單說明：

柒、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

捌、聲明事項

☐ 本案係符合專利法第二十條第一項☐第一款但書或☐第二款但書規定之期間，其日期爲：_____

☐ 本案已向下列國家（地區）申請專利，申請日期及案號資料如下：

【格式請依：申請國家（地區）；申請日期；申請案號　順序註記】

1._____

2._____

3._____

☐ 主張專利法第二十四條第一項優先權：

【格式請依：受理國家（地區）；日期；案號　順序註記】

1._____

2._____

3._____

4._____

5._____

6._____

7._____

8._____

9._____

10._____

☐ 主張專利法第二十五條之一第一項優先權：

【格式請依：申請日；申請案號　順序註記】

1._____

2._____

3._____

☐ 主張專利法第二十六條微生物：

☐國內微生物　【格式請依：寄存機構；日期；號碼　順序註記】

1._____

2._____

3._____

☐國外微生物　【格式請依：寄存國名；機構；日期；號碼　順序註記】

1._____

2._____

3._____

☐熟習該項技術者易於獲得，不須寄存。

玖、發明說明

（發明說明應敘明：發明所屬之技術領域、先前技術、內容、實施方式及圖式簡單說明）

（一）、【發明所屬之技術領域】

本發明係有關一種語音辨認之訓練方法，特別是關於一種應用於雜訊環境中具有高辨識率之語音模型訓練方法。

（二）、【先前技術】

隨著電子技術的發達，電子產品已與資訊、通訊二項產品的技術結合在一起，並利用網路將它們連接起來，創造一自動化之生活環境，使生活及工作更加便利。其中，使用者使用不同的通訊產品，在不同環境來使用語音辨認器，然而多樣性的雜訊環境會破壞語音辨認裝置的辨識率。

語音辨認通常分為二個階段，一為訓練階段，二為辨認階段。在訓練階段，首先係收集不同聲音且以統計之方式產生一語音模型，而後將此語音模型套入學習程序，以使語音辨認裝置具備學習能力，當使用一段時間反覆訓練後，加上比對辨認技術，達到提昇語音辨識能力之作用。因此，訓練模型所運用之訓練方法將影響語音辨認裝置之辨識能力甚深。

習知語音訓練法主要有鑑別式訓練法(Discriminative training techniques)及強健式訓練法

(Robust Environment-effects Suppression Training，REST)，鑑別式訓練法係藉由統計方式將具有一定相似度容易混淆之語音訊號加以統計，訓練時會考慮具有容易混淆的語音訓練資料從而產生鑑別度高之模型，此訓練法在安靜環境下對於乾淨聲音之學習效果較好，但在雜訊環境下易受環境中之雜訊影響而表現不佳；除了這個缺點，在雜訊環境下實施鑑別式訓練法，所產生的語音模型有過度吻合(over-fitting)及缺乏普遍化(generalization)能力，也就是說此鑑別式模型已經調適成適合於某種雜訊環境之模型，但當測試的環境稍微改變，則辨認效果就大幅下降。另一方面，強健式訓練法係除了統計具相似度之語音訊號之外，且將環境效應壓抑，以加強語音辨認之強健能力，此種訓練法雖具有強健力之特性，但語音模型鑑別力則表現較鑑別式訓練法為差。

因此，本發明係針對上述之問題，提出一種應用於語音辨認之語音模型訓練方法，以便在雜訊環境下能同時兼具鑑別力及強健力。

(三)、【發明內容】

本發明之主要目的，係在提供一種應用於語音辨認之語音模型訓練方法，其係先以強健式訓練法將輸入語音中

之環境因素分離，再利用鑑別式訓練法針對乾淨之聲音進行訓練，藉由整合鑑別式及強健式訓練法以使得到之語音訓練模型，同時兼具有強健能力及鑑別能力，以克服習知二者無法兼具之缺失，進而提高辨識率。

本發明之另一目的，係在提供一種應用於語音辨認之語音模型訓練方法，以適用於雜訊環境的補償辨認，達到提高雜訊環境中語音辨識率之功效。

本發明之再一目的，係在將輸入語音中之各聲音效應單獨分離，使各失真因素個別分開，以達到精準之環境效應調控。

根據本發明，一種應用於語音辨認之語音模型訓練方法係包括下列步驟：將輸入語音分離成為一乾淨聲音之密實語音模型及一環境因素模型；接著，根據該環境因素模型將輸入語音中之環境因素濾除而得到一語音訊號；再將此語音訊號與密實語音模型利用鑑別式訓練法演算而得到一高鑑別度且密實的語音訓練模型，以提供語音辨認裝置進行後續之語音辨認處理。

底下藉由具體實施例配合所附的圖式詳加說明，當更容易瞭解本發明之目的、技術內容、特點及其所達成之功效。

(四)、【實施方式】

本發明之語音模型訓練方法係先利用強健式訓練法將輸入語音分離且模型化成為密實語音模型(compact model)及環境因素模型，以便使密實語音模型做為一強健式種子模型而進行模型補償，並藉由鑑別式訓練法演算而得到一高鑑別度之語音訓練模型，以提供語音辨認裝置進行後續之語音辨認處理。

第一(a)圖及第一(b)圖為本發明於建立語音模型訓練方法之架構示意圖，首先，如第一(a)圖所示，利用強健式訓練法(Robust Environment-effects Suppression Training，REST)[1]將輸入語音 Z 計算而模型化分離出一密實語音模型 Λ_x 及一環境因素模型 Λ_e，環境因素模型 Λ_e 之訊號係包括通道訊號及雜訊，通道訊號常見者包括有麥克風效應或語者偏差值(speaker bias)；而後如第一(b)圖所示，利用該環境因素模型 Λ_e 壓抑輸入語音 Z 之環境因素而得到一語音訊號 X，此濾除環境因素之步驟通常係利用一濾波器進行；最後，利用鑑別式訓練法中之通用型或然性下降訓練法(generalized probabilistic descent，GPD)將已壓抑環境因素之語音訊號 X 套入於密實語音模型 Λ_x 中，經演算後即得到一高鑑別度且密實之語音模型 Λ_x'。

在利用本發明之演算法得到上述高鑑別度且密實之語

音模型Λ_x'後，在應用於語音辨認裝置的辨認階段中，係運用一平行模型結合方法(parallel model combination，PMC)及訊號偏差補償(signal bias compensation，SBC)式辨認法，通常稱為PMC-SBC法(參附件一)，對語音模型Λ_x'進行補償以符合目前運作環境，而後進行辨認程序。此'PMC-SBC方法如下：首先，藉由比較類神經網路(Recurrent Neural Network，RNN)之非語音輸出與一預定之臨界值(threshold)，以偵測出非語音音框(non-speech frame)，且將此非語音音框使用於計算線上(on-line)雜訊模型；而後利用狀態式維納濾波方法(state-based Wiener filtering method，其係利用平穩隨機過程的相關特性和頻譜特性對混有噪聲的信號進行濾波的方法)將輸入語音中之第 r 個語句(utterance)$Z^{(r)}$進行處理而得到增強語音訊號；而後將該增強訊號之語句$Z^{(r)}$轉換為一倒頻譜頻域(cepstrum domain)以藉由SBC方法估算通道偏差值，在此SBC法中，係先使用代碼本(codebook)來將該增強語句$Z^{(r)}$之特徵向量進行轉碼(encoding)，再計算平均轉碼剩餘值(encoding residuals)，其中代碼本係藉由收集密實語音Λ_x中混合組成的平均向量而形成；而後以此通道偏差值將所有語音模型Λ_x轉換為偏差補償式語音模型，接著，更進一步地利用PMC方法且使用線上雜訊模型(on-line noise

model)將被該些偏差補償式語音模型轉換為雜訊(noise-)及偏差(bias-)補償式語音模型；該等雜訊及偏差償式語音模型即可使用於後續之輸入語句$Z^{(r)}$的辨認工作。

本發明之語音模型訓練方法係可應用於具有語音辨認器之裝置，如汽車語音辨認器、個人數位助理(PDA)語音辨認器及電話/手機語音辨認器等裝置。

因此，本發明先藉由強健式訓練法將輸入語音中之雜訊分離，再利用鑑別式訓練法針對乾淨之聲音進行訓練，藉由整合鑑別式及強健式訓練法以使得到之密實語音訓練模型，不僅同時兼具有強健能力及鑑別能力，且更適用於雜訊環境的補償辨認；另外，由於本發明之學習方法可將輸入語音中之各聲音效應單獨分離，因此可將各失真因素個別分開，可應用於選擇性的環境效應訊號調控，如環境因素對語音之調控或語者模型之調適上。

至此，本發明之演算法的精神已說明完畢，以下特以一具體理論推導來詳細驗證說明本發明之演算法。本發明之演算法係為鑑別及強健式訓練方法(Discriminative REST，以下簡稱D-REST)，係基於在一假設之雜訊模型中，由均勻且乾淨之聲音 $X^{(r)}$經過此雜訊模型而得到$Z^{(r)}$。其中，$Z^{(r)}$代表第 r 個語句(utterance)之聲音特徵向量。考慮一組鑑別函數$\{g_i,\ i=1,2...,M\}$及$Z^{(r)}$之環境補償聲音

HMM模型$\Lambda_z^{(r)}$，定義：

$$g_i(Z^{(r)};\Lambda_z^{(r)}) \equiv \log[\Pr(Z^{(r)},U_i^{(r)}|\Lambda_z^{(r)})]$$
$$= \log[\Pr(Z^{(r)},U_i^{(r)}|\Lambda_x \otimes \Lambda_e)] \quad (1)$$

其中，$U_i^{(r)}$為$Z^{(r)}$對$\Lambda_z^{(r)}$之第i個隱藏式馬可夫模型(Hidden Markov Model，HMM)之最大相似狀態之組態；Λ_x代表環境因素壓抑之HMMs模型，亦即密實語音模型(compact model)，而Λ_e係為環境因素模型；⊗符號代表模型補償(model compensation)之運算符號，其亦運用在辨認過程中。

本發明D-REST演算法之目標是根據鑑別函數g_i來估算Λ_x及Λ_e模型，且使Λ_x做為一強健及鑑別式之種子模型，以做為模型補償時之雜訊環境聲音辨認。

D-REST演算法之第一步驟係同時計算出密實語音模型Λ_x及環境因素模型Λ_e。假設在每一語音中，環境因素包括一常見之通道b及一附加雜訊n。令$\Lambda_e \equiv \{\Lambda_n^{(r)},b^{(r)}\}_{r=1,\ldots,R}$，其係代表在整個訓練資料中之環境因素模型，其中$b^{(r)}$及$\Lambda_n^{(r)}$分別表示在第$r$訓練語句中之訊號偏差及雜訊模型(noise model)，根據最大相似度準則(maximum likelihood criterion)，利用所給之$\{Z^{(r)}\}_{r=1,\ldots,R}$同時計算出$\Lambda_x$及$\Lambda_e$，係透過下列公式來獲得：

$$(\Lambda_x, \Lambda_e) = \arg \max_{(\overline{\Lambda}_x, \overline{\Lambda}_e)} \Pr\left(\{Z^{(r)}\}_{r=1,\ldots,R} \middle| \overline{\Lambda}_x, \overline{\Lambda}_e\right) \qquad (2)$$

在反覆的訓練過程中，利用強健式訓練法(REST)來相繼地進行方程式(1)之運算，包括下列三個操作流程：(1)藉由使用當下的$\{\Lambda_x, \Lambda_e\}$計算值來形成補償的HMMs $\Lambda_z^{(r)}$值，且利用此$\Lambda_z^{(r)}$值來最佳化地分割該訓練的語調$Z^{(r)}$；(2)根據分割結果計算$\Lambda_n^{(r)}$來強化不利聲音(adverse speech)$Z^{(r)}$，以獲得$Y^{(r)}$，而後計算$b^{(r)}$且更進一步地強化$Y^{(r)}$聲音以得到$X^{(r)}$；(3)利用強化的$\{X^{(r)}\}_{r=1,\ldots,R}$聲音來更新當下的HMMs模型$\Lambda_x$。

在訓練過程中，由於涉及環境因素補償之運算，因此可預期將會產生較佳之參考語音模型以提供強健式辨識方法。再者，Λ_x及Λ_e之分離模型化可使訓練過程集中在語音的音素變化(phonetic variation)之模型化上，而排除來自於環境因素之不當影響。

D-REST 演算法之第二步驟係在表現最小錯誤辨識率(minimum classification error，MCE)的鑑別式訓練法，其係根據上述利用環境補償聲音 HMM 模型$\Lambda_z^{(r)}$和觀察語音 Z 而演算得之。在此係採用鑑別式訓練法中的部分式通用型或然性下降訓練法(segmental GPD)(參附件二)，其係使用下列計算式來量測$Z^{(r)}$：

$$d_i(Z^{(r)} | \Lambda_z^{(r)}) = -g_i(Z^{(r)}; \Lambda_z^{(r)}) + g_k(Z^{(r)}; \Lambda_z^{(r)}) \qquad (3)$$

其中，$k = \text{argmax}_{j,\, j\neq i} \Pr\left(Z^{(r)}, U_j^{(r)} \middle| \Lambda_z^{(r)}\right)$；基於上述運算式且假設$\sum_{z,j,q}^{(r)} = \sum_{x,j,q}$和狀態式維納濾波方法為PMC的反運算(參附件三)，則在運算式(1)中之$\Pr\left(Z^{(r)}, U_i^{(r)} \middle| \Lambda_z^{(r)}\right)$項可再被寫為：

$$\Pr\left(Z^{(r)}, U_i^{(r)} \middle| \Lambda_z^{(r)}\right) = \Pr\left(Z^{(r)}, U_i^{(r)} \middle| \left\{\mu_{x,j,q}^{(r)} + b^{(r)} - h_j, \Sigma_{z,j,q}^{(r)}\right\}\right) \quad (4)$$

$$= \Pr\left(X^{(r)}, U_i^{(r)} \middle| \left\{\mu_{x,j,q}^{(r)}, \Sigma_{x,j,q}^{(r)}\right\}\right)$$

$$= \Pr\left(X^{(r)}, U_i^{(r)} \middle| \Lambda_x\right)$$

因此，方程式(3)可表示為：

$$d_i\left(Z^{(r)} \middle| \Lambda_z^{(r)}\right) = d_i\left(X^{(r)} \middle| \Lambda_x\right) \quad (5)$$

由方程式(5)顯示，採用MCE訓練法對語音Z與環境補償聲音HMM模型$\Lambda_z^{(r)}$做訓練，等同於採用MCE訓練法對環境因素已壓抑語音X且給定之環境因素模型Λ_e做訓練。

因此，經由上述語音模型訓練方法之推演可得到一高鑑別度且密實的語音訓練模型，以下將以二具體實施例來驗證說明本發明之作用及功效。第一實施例請參閱第二圖所示，係為將本發明之D-REST訓練方法及習知鑑別式訓練方法(GPD)、強健式訓練方法(REST)應用在GSM傳輸通道之汽車雜訊環境中，在處於不同訊雜比環境下對於語音辨識錯誤率之比較，其中，對照組係為沒有任何雜訊模型補償之傳統HMM辨認方法。由測試結果可清楚得知，無論係在乾淨之聲音中，或是在訊雜比僅為3之高雜訊環境中，當汽車內之語音辨識裝置使用本發明之D-REST語音模型訓練方法

時，皆具有最低之錯誤辨識率，因而可達到最佳之辨識效果。

第三圖所示為另一具體實施例，其測試條件及標的和第一實施例相同，但訓練語料的汽車雜訊和測試語料的汽車雜訊類別不同。由測試結果可清楚得知，使用本發明之D-REST語音模型訓練方法時，在不同訊雜比下皆具有最低之錯誤辨識率;而採用鑑別式訓練方法(GPD)的結果則反而比對照組更差，這是因為所產生的語音模型有過度吻合(over-fitting)及缺乏普遍化(generalization)的問題，因此當測試的環境稍微改變，則辨認效果就下降。

以上所述係藉由實施例說明本發明之特點，其目的在使熟習該技術者能瞭解本發明之內容並據以實施，而非限定本發明之專利範圍，故，凡其他未脫離本發明所揭示之精神所完成之等效修飾或修改，仍應包含在以下所述之申請專利範圍中。

(五)、【圖式簡單說明】

圖式說明：

第一(a)圖至第一(b)圖為本發明於建立語音模型訓練方法之架構示意圖。

第二圖為具體使用本發明之訓練方法與習知訓練方法之辨

識結果比較示意圖。

第三圖為具體使用本發明之訓練方法與習知訓練方法之另一辨識結果比較示意圖。

申請專利範圍

1. 一種應用於語音辨認之語音模型訓練方法，包括下列步驟：

將輸入語音分離成為一乾淨聲音之密實語音模型及一環境因素模型；

根據該環境因素模型將該輸入語音中之環境因素濾除而得到一語音訊號；以及

將該語音訊號套入該密實語音模型中，且利用鑑別式訓練法演算而得到一語音訓練模型，以提供語音辨認裝置進行後續之語音辨認處理。

2. 如申請專利範圍第 1 項所述之語音模型訓練方法，其中，該環境因素模型之訊號係包括通道訊號及雜訊。

3. 如申請專利範圍第 2 項所述之語音模型訓練方法，其中，該通道訊號係包括麥克風通道效應。

4. 如申請專利範圍第 2 項所述之語音模型訓練方法，其中，該通道訊號係包括語者偏差值(speaker bias)。

5. 如申請專利範圍第 1 項所述之語音模型訓練方法，其中，該鑑別式訓練法係通用型或然性下降訓練法(generalized probabilistic descent，GPD)。

6. 如申請專利範圍第 1 項所述之語音模型訓練方法，其中，分離該輸入語音之步驟係藉由比較類神經網路非語音輸出與一預定之閥值以偵測出非語音音框，且將此非語音

音框套用於計算線上(on-line)雜訊模型上。

7.如申請專利範圍第 1 項所述之語音模型訓練方法，其中，濾除該環境因素之步驟係利用一濾波器進行。

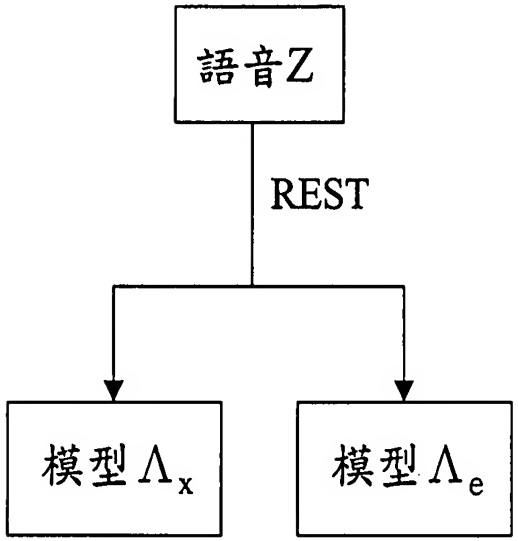
8.如申請專利範圍第 1 項所述之語音模型訓練方法，其中，濾除該環境因素之步驟更包括：

利用狀態式維納濾波方法(state-based Wiener filtering method)處理該輸入語音以使該密實語音模型進而成為一增強狀態組態之語音；

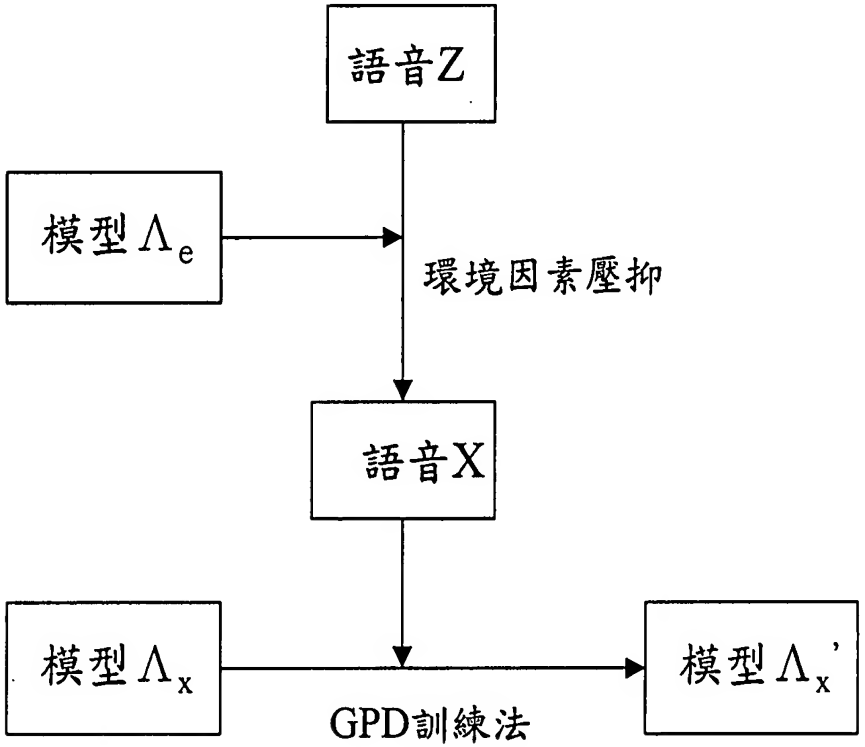
將該增強狀態組態之語音轉換為一倒頻譜頻域(Cepstrum Domain)，以藉由訊號偏差補償(signal bias compensation)方法估算偏差值，而將該密實語音模型轉換為偏壓補償式語音模型；以及

利用平行模型結合法(parallel model combination)且使用一線上雜訊模型將被該偏壓補償式語音模型轉換為雜訊及偏壓補償式語音模型。

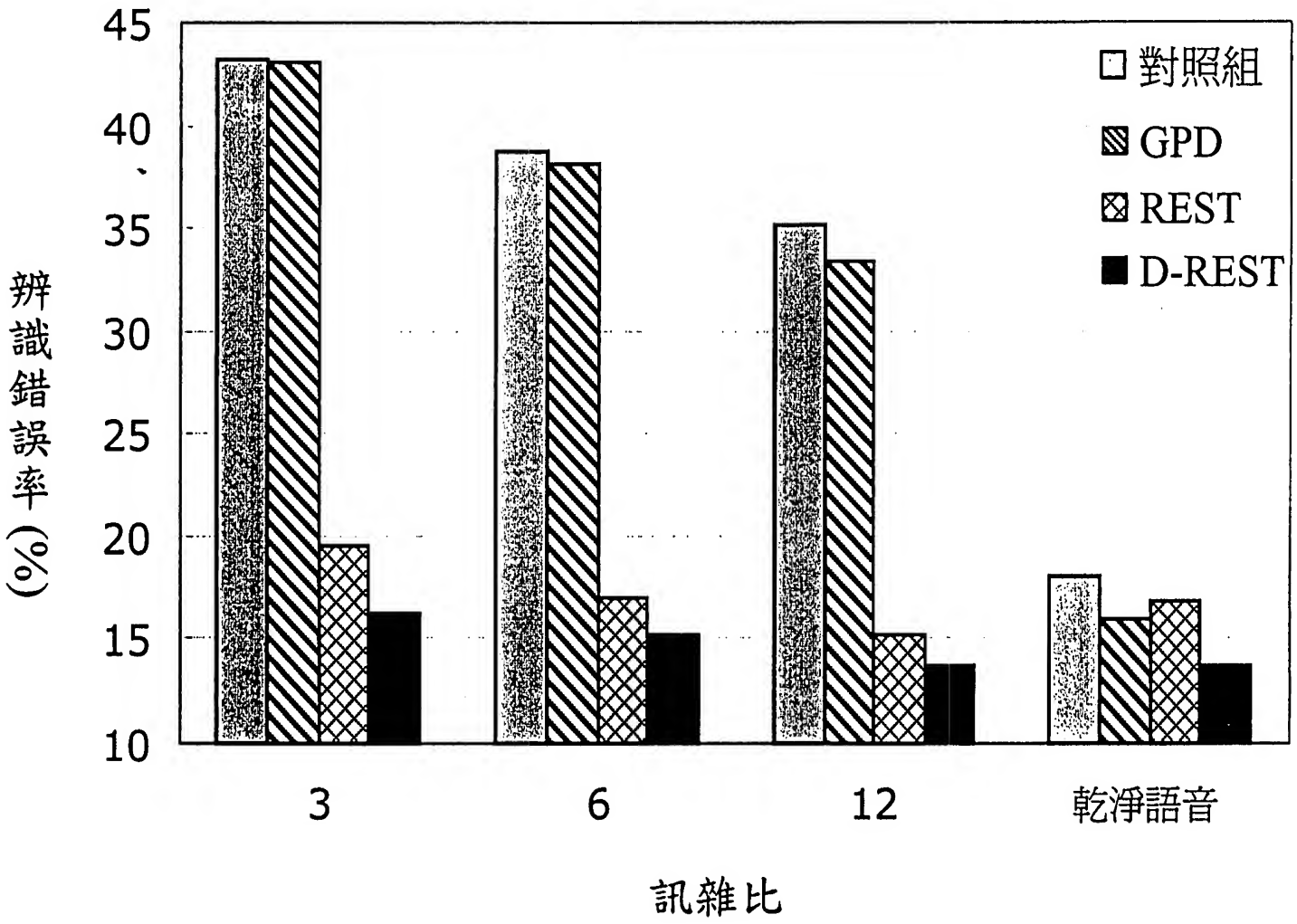
9.如申請專利範圍第 8 項所述之語音模型訓練方法，其中，在該訊號偏壓補償方法中，係先使用代碼本將該增強狀態組態之語音的特徵向量進行轉碼，再計算平均轉碼剩餘值，其中代碼本係藉由收集該等密實語音模型中混合組成的平均向量而形成。


語音Z
REST
模型Λ_x
模型Λ_e

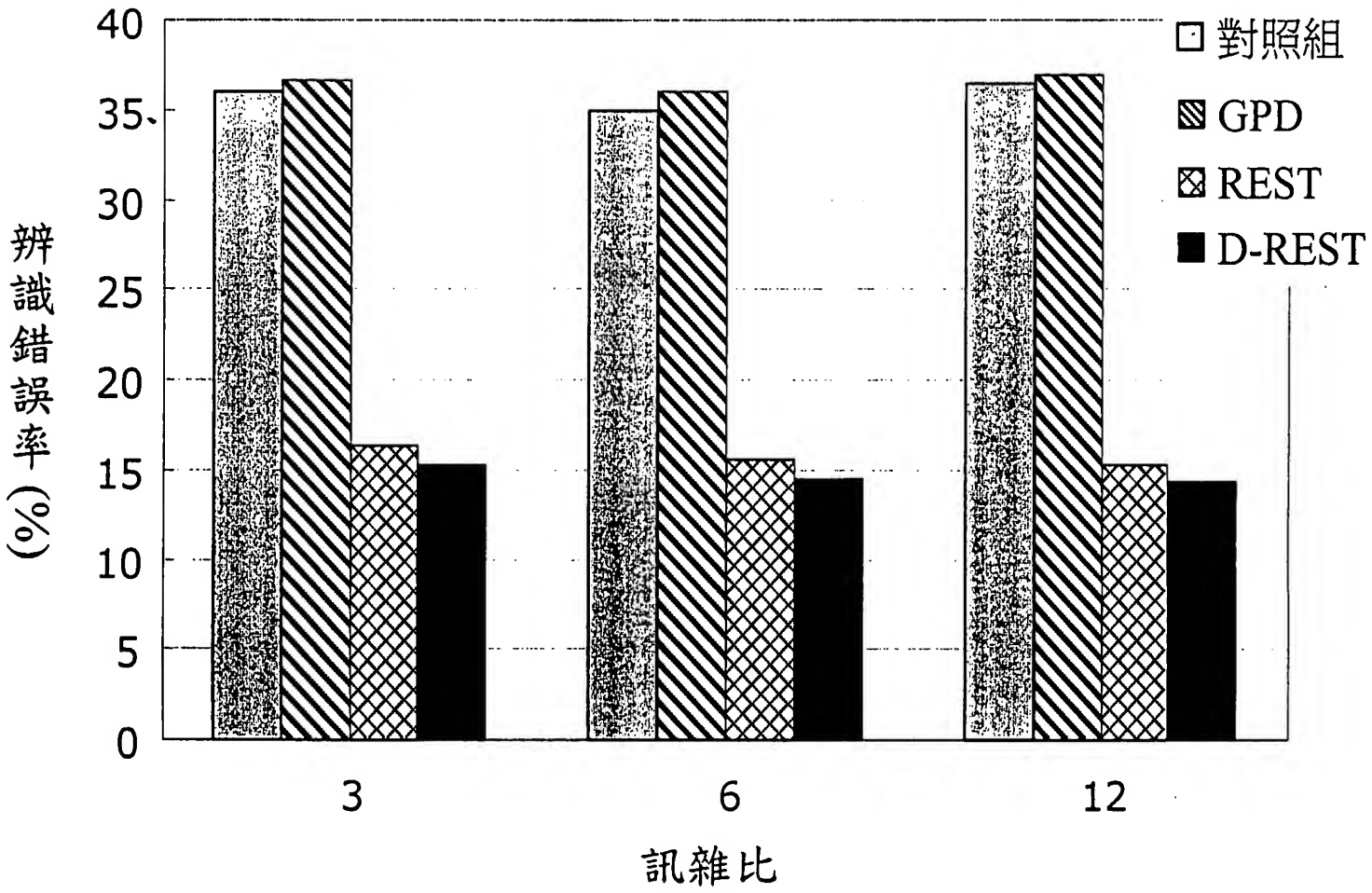
第一(a)圖


語音Z
模型Λ_e
環境因素壓抑
語音X
模型Λ_x
模型Λ_x'
GPD訓練法

第一(b)圖



第二圖



第三圖

ELSEVIER

Speech Communication 30 (2000) 273–293

SPEECH COMMUNICATION

www.elsevier.nl/locate/specom

A robust training algorithm for adverse speech recognition

Wei-Tyng Hong [a,*], Sin-Horng Chen [b]

[a] *E000/CCL, Industrial Technology Research Institute, Chutung, Hsinchu, Taiwan, ROC*
[b] *Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan, ROC*

Received 10 November 1998

Abstract

In this paper, a new robust training algorithm is proposed for the generation of a set of bias-removed, noise-suppressed reference speech HMM models in adverse environment suffering from both channel bias and additive noise. Its main idea is to incorporate a signal bias-compensation operation and a PMC noise-compensation operation into its iterative training process. This makes the resulting speech HMM models more suitable to the given robust speech recognition method using the same signal bias-compensation and PMC noise-compensation operations in the recognition process. Experimental results showed that the speech HMM models it generated outperformed both the clean-speech HMM models and those generated by the conventional *k*-means algorithm for two adverse Mandarin speech recognition tasks. So it is a promising robust training algorithm.

Keywords: Robust training algorithm; PMC noise-compensation; Signal bias-compensation; Mandarin speech recognition

1. Introduction

Background noise and channel bias are the two major interference factors that seriously degrade the performances of speech recognizers operating in adverse environments such as telephone speech through public switching network. Recently, IBM built an HMM-based Mandarin telephone speech recognition system using a large telephone speech database called 'Mandarin call home database' (Liu et al., 1996). The vocabulary contained about 44 000 words. The word and syllable error rates were, respectively, 70.5% and 58.7%, which were much worse than those achieved in microphone-speech recognition (Lee and Juang, 1996). In the past, many studies have been devoted to the field of robust speech recognition for adverse environment (Juang, 1991; Furui, 1992; Gong, 1995; Junqua and Haton, 1996). Major efforts of those studies were put on developing robust recognition algorithms to compensate or to eliminate noise/channel effect based on a given set of reference speech models trained usually in clean-speech environment. In the non-linear noise subtraction method (Lockwood and Boudy, 1992; Mokbel and Chollet, 1995), a noise model was first estimated from the non-speech precursor of the testing utterance and then subtracted from the speech part in linear spectrum domain in order to obtain noise-suppressed features to be recognized using the clean-speech reference models. In (Acero and Stern, 1990, 1991), the CDCN

* Corresponding author.
E-mail addresses: jfhong@taiwan.com (W.-T. Hong), schen@cc.nctu.edu.tw (S.-H. Chen).

0167-6393/00/$ - see front matter
PII: S0167-6393(99)00057-6

(codeword-dependent cepstral normalization) algorithm was proposed to estimate equalization vectors for the best transformation, in the maximum likelihood sense, from the universal codebook into the testing acoustic space in order to eliminating both the noise and channel effects. In the RASTA method (Hermansky and Morgan, 1994), a filter was used to eliminate the speaker/channel bias for obtaining bias-removed recognition features. In the parallel-model-combination (PMC) method (Gales and Young, 1996), clean-speech HMM models were combined with the current noise model to form noise-compensated composite HMM models for recognizing noisy speech. In the state-based Wiener filtering method (Hansen and Clements, 1991; Ephraim, 1992; Vaseghi and Milner, 1997), a two-stage recognition method was used. It first used the Viterbi algorithm in the first stage to find the best state sequence for the input testing noisy speech, and then applied state-based Wiener filtering to estimate the clean-speech and recognized it using the clean-speech HMM models in the second stage. In (Zhao, 1996), a two-step procedure was employed to detect a spectral bias vector for the input testing utterance by using Gaussian distributed phone models. It then removed the estimated bias vector from the testing utterance for recognition. In the stochastic matching algorithm (Sankar and Lee 1996; Lee, 1998), the parameters of mapping functions between the testing speech and reference HMM models were estimated iteratively using the expectation maximization (EM) algorithm (Dempster et al., 1977). In (Minami and Furui, 1996), an integrated method for adapting HMM models to additive noise and channel distortion was proposed. This method first estimated the signal-to-noise ratio by maximizing the likelihood of the PMC-compensated HMM models to the input speech, and then estimated the cepstral bias by the Sankar's method (Sankar and Lee, 1996). The procedure is iteratively applied until a convergence is reached.

Apart from the above-mentioned main research stream, the robust training issue is also important for adverse speech recognition when the clean-speech reference models are not available. Its main concern is to train a set of robust reference speech models directly from a database collected in adverse environment for adverse speech recognition. The issue is important because the set of reference speech models obtained by the conventional segmental k-means algorithm (Juang and Rabiner, 1990) is usually not robust. This is mainly owing to the high variability on the characteristics of the training speech signals collected in the adverse environment. For example, a training data set collected from telephone calls through the public switching network will suffer diverse recording conditions caused by different background noises, different types of transducers, different telephone channels, etc. This will make speech patterns distribute more widely in the feature space so as to overlap to each other more seriously and cause the trained speech models degrade on their discrimination capabilities.

In the past, many robust training algorithms have been proposed. In the signal bias removal (SBR) algorithm (Rahim and Juang, 1996), a codebook-based iterative signal bias removing technique was performed on both the training and testing phases for minimizing the channel-induced variations. In (Anastaskos et al., 1997), the speaker-specific characteristics were first modeled by a linear-regressive transformation between the speaker-independent models and the speaker-dependent models. A speaker-adaptive training algorithm designed basing on the EM algorithm was then employed to iteratively estimate the parameters of the transformation and the compact speaker-normalized HMM models. In (Gong, 1997), a source normalization training algorithm, which modeled the environmental corruption as a form of linear transformation, was proposed to estimate the HMM models. The noise and channel effects were modeled implicitly in the linear transformation. In the testing stage, the MLLR adaptation (Gales and Woodland, 1996) was applied to estimate the state-dependent transformation matrices and the bias terms for recognition. Those training algorithms have been shown to be effective on removing the channel biases and/or the speaker variations. However, the noise effect is still seldom considered in the robust training issue.

In this study, we are interested in the robust training issue with both the signal bias and noise effects being considered. A robust training algorithm, referred to as the robust environment-effects suppression training (REST) algorithm, is proposed. The design goal of the REST algorithm is twofold. One is to countervail the large variability of the corrupted training samples for obtaining a set of compact reference

speech HMM models with both signal bias and noise being suppressed. The other is to make the generated compact reference speech HMM models better for a given robust speech recognition method. The REST algorithm is an iterative training procedure that sequentially optimizes the following three operations: parameter estimation for environment characterization, environment-effect compensation for speech segmentation, and environment-effect suppression for HMM model re-estimation. The parameter estimation for environment characterization is to detect the signal bias and to estimate the noise statistics for each training utterance. It assumes that each utterance has its own environmental characteristics. Based on an assumed environment contamination model, the environment-effect compensation uses the estimated environment characterization parameters to adapt the HMM models to match with the current training utterance for optimal segmentation. Using the segmentation results and the same environment contamination model, the environment-effect suppression is to remove the signal bias and the noise out of the corrupted speech for updating the HMM models. Owing to the involvement of the environment-effect compensation operation in the training process of the REST algorithm, we expect that it will generate better reference speech HMM models for the robust recognition method which employs the same environment-effect compensation operation in the recognition process. This is especially true for the case when the environment-effect compensation operation is not perfect due either to the non-existence of a perfect one or to the use of an inaccurate environment contamination model in its derivation.

The organization of the paper is stated as follows. Section 2 presents the proposed REST algorithm in detail. Section 3 describes the robust speech recognition method using the reference speech HMM models generated by the REST algorithm. Effectiveness of the REST algorithm is evaluated by simulations discussed in Section 4. Some conclusions are given in Section 5.

2. The REST algorithm

The proposed REST training algorithm consists of an iterative procedure which sequentially performs the following three steps:

1. optimally segment each training utterance by using the environment-compensated HMM models,
2. estimate the environment characteristics and enhance the speech by eliminating the noise using the state-based Wiener filtering method and by removing the signal bias using the SBR method, and
3. re-estimate the speech HMM models.

Operations performed in these three steps are derived based on a presumed environment contamination model. A schematic diagram of the model is displayed in Fig. 1. It assumes that, for each utterance, the observed speech z is generated from the clean speech x by corrupting first with a convolutional channel b and then with an additive noise n. Here b is assumed to be time-invariant and n is stationary throughout the utterance. In linear spectrum domain, the model can be expressed by

$$y_t(f) = b(f) \times x_t(f), \tag{1a}$$

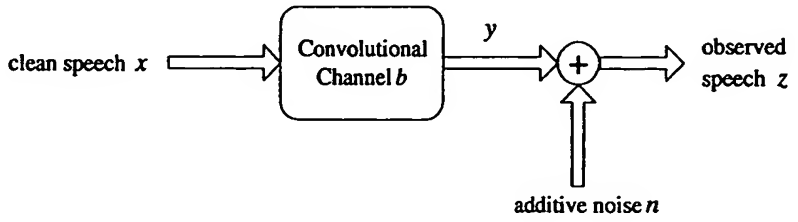


Fig. 1. A schematic diagram of the environment contamination model.

$$z_t(f) = y_t(f) + n_t(f), \tag{1b}$$

where the subscript t denotes the frame index and $y_t(f)$ is an intermediate signal showing the corruption of the clean-speech with the channel bias only. We can also express the relation of x and y in cepstrum domain by

$$y_t(m) = b(m) + x_t(m), \tag{1c}$$

where m denotes the order of cepstral coefficient. Obviously, it is troublesome to directly estimate the original clean speech x in either linear spectrum domain or cepstrum domain when both noise interference and channel distortion exist. We had better, as suggested by above formulations, to separately deal with the channel distortion in cepstrum domain and the noise interference in linear spectrum domain. In the following discussions, we specify a signal in linear spectrum domain and in cepstrum domain by attaching it with parameters f and m, respectively.

The REST algorithm is derived as follows. Assume that the training data set contains R utterances. Let $\Lambda_e \equiv \{\Lambda_n^{(r)}, b^{(r)}\}_{r=1,\ldots,R}$ denote the set of environmental interference models of the whole training data set, where $b^{(r)}$ and $\Lambda_n^{(r)} = \{\mu_n^{(r)}, \Sigma_n^{(r)}\}$ are, respectively, the signal bias and the noise model of the rth training utterance; $\mu_n^{(r)}$ and $\Sigma_n^{(r)}$ are the mean vector and covariance matrix of $\Lambda_n^{(r)}$. Let $Z^{(r)} = (z_1^{(r)}, \ldots, z_{T_r}^{(r)})$ and $X^{(r)} = (x_1^{(r)}, \ldots, x_{T_r}^{(r)})$ be, respectively, the observed and clean-speech feature vector sequence of the rth utterance, and Λ_x denote the set of environment-effect normalized speech HMM models that we want to generate. Based on the maximum likelihood criterion, the goal of an ideal robust training algorithm is to jointly estimate Λ_x and Λ_e with given $\{Z^{(r)}\}_{r=1,\ldots,R}$ by

$$(\Lambda_x^*, \Lambda_e^*) = \arg\max_{(\Lambda_x, \Lambda_e)} L\left(\{Z^{(r)}\}_{r=1,\ldots,R} \middle| \Lambda_x, \Lambda_e\right), \tag{2}$$

where $L(\cdot)$ is the likelihood function of the observation sequence $Z^{(r)}$ given the parameter set of (Λ_x, Λ_e). But, due to the fact that it is generally difficult to derive a close form solution for the above joint maximization problem, we therefore use a three-step iterative training procedure in the REST algorithm to obtain a sub-optimal solution. The three steps are:

1. Form the environment-compensated speech HMM models $\Lambda_z^{(r)}$ by using the current (Λ_x, Λ_e) and use it to optimally segment the training utterance $Z^{(r)}$.
2. Based on the segmentation result, estimate $\Lambda_n^{(r)}$ and enhance the adverse speech $Z^{(r)}$ to obtain $Y^{(r)}$ by the state-based Wiener filtering method; and then, estimate $b^{(r)}$ and further enhance the speech $Y^{(r)}$ to obtain $X^{(r)}$ by the SBR method.
3. Update the current speech HMM models Λ_x using the enhanced speech $\{X^{(r)}\}_{r=1,\ldots,R}$. We discuss these three steps in more detail as follows.

The first step of the REST algorithm is to optimally segment each training utterance using the current speech HMM models $\Lambda_{x,k-1}$ and the environmental interference model $\Lambda_{e,k-1}$ given by the previous iteration, where the subscript k denotes the index of iteration. The task can be accomplished, based on the maximum likelihood criterion, by solving the following optimization problem to find the best state sequence $U_k^{(r)} = (u_{1,k}^{(r)}, \ldots, u_{T_r,k}^{(r)})$ and the best mixture component sequence $V_k^{(r)} = (v_{1,k}^{(r)}, \ldots, v_{T_r,k}^{(r)})$ of the optimal segmentation:

$$\begin{aligned}(U_k^{(r)}, V_k^{(r)}) &= \arg\max_{(U^{(r)}, V^{(r)})} \Pr\left(Z^{(r)}, U^{(r)}, V^{(r)} \middle| \Lambda_{x,k-1}, \Lambda_{e,k-1}\right) \\ &= \arg\max_{((u_1^{(r)},\ldots,u_{T_r}^{(r)}),(v_1^{(r)},\ldots,v_{T_r}^{(r)}))} \left\{\prod_{t=1}^{T_r} a_{u_{t-1}^{(r)}, u_t^{(r)}} \Pr\left(z_t^{(r)} \middle| u_t^{(r)}, v_t^{(r)}, \Lambda_{z,k-1}^{(r)}\right)\right\},\end{aligned} \tag{3}$$

where $a_{i,j}$ denotes the transition probability from state i to state j. Eq. (3) is solved in this study by first forming the environment-compensated speech HMM models $\Lambda_{z,k-1}^{(r)}$ using $\Lambda_{x,k-1}$ and $\Lambda_{e,k-1}$, and then using the Viterbi search to simultaneously find $U_k^{(r)}$ and $V_k^{(r)}$. The formation of $\Lambda_{z,k-1}^{(r)}$ from $\Lambda_{x,k-1}$ and $\Lambda_{e,k-1}$ is based on the assumed environment contamination model defined in Eqs. (1b) and (1c), and realized by the following two sub-steps:

(1.1) Calculate $\Lambda_{y,k-1}^{(r)}$ in cepstrum domain by

$$\mu_{y,j,q,k-1}^{(r)}(m) = \mu_{x,j,q,k-1}(m) + b_{k-1}^{(r)}(m), \tag{4a}$$

$$\Sigma_{y,j,q,k-1}^{(r)}(m) = \Sigma_{x,j,q,k-1}(m), \tag{4b}$$

where $\mu_{y,j,q,k-1}^{(r)}(m)$ and $\Sigma_{y,j,q,k-1}^{(r)}(m)$ are, respectively, the mean vector and covariance matrix of the qth Gaussian mixture in the jth state of $\Lambda_{y,k-1}^{(r)}$, and $b_{k-1}^{(r)}(m)$ is the bias vector given in $\Lambda_{e,k-1}$.

(1.2) Use the PMC method to form $\Lambda_{z,k-1}^{(r)}$ by first transforming $\Lambda_{y,k-1}^{(r)}$ from cepstrum domain to linear spectrum domain, then combining it with $\Lambda_{n,k-1}^{(r)}$ in linear spectrum domain, and lastly transforming the result back to cepstrum domain.

The second step of the REST algorithm is to enhance the adverse speech by first suppressing the noise using the state-based Wiener filtering method (Hansen and Clements, 1991; Ephraim, 1992; Vaseghi and Milner 1997) and by then removing the signal bias by the SBR method (Rahim and Juang, 1996). It consists of the following two sub-steps:

(2.1) Noise suppression: Given the segmentation information $U_k^{(r)}$, estimate the noise model $\Lambda_{n,k}^{(r)}$ and eliminate it from the input adverse speech $z_t^{(r)}(f)$, in linear-spectrum domain, by the state-based Wiener filtering method to obtain the intermediate signal $y_{t,k}^{(r)}(f)$. The noise model $\Lambda_{n,k}^{(r)}$ and its average power spectrum density $P_{n,k}^{(r)}(f)$ of the rth utterance are re-estimated from the non-speech frames by

$$\mu_{n,k}^{(r)}(m) = \frac{\sum_{t=1}^{T_r} z_t^{(r)}(m) \times I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)}{\sum_{t=1}^{T_r} I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)}, \tag{5a}$$

$$\Sigma_{n,k}^{(r)}(m) = \frac{\sum_{t=1}^{T_r} \left(z_t^{(r)}(m)\right)^2 \times I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)}{\sum_{t=1}^{T_r} I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)} - \left(\mu_{n,k}^{(r)}(m)\right)^2, \tag{5b}$$

$$P_{n,k}^{(r)}(f) = \frac{\sum_{t=1}^{T_r} \hat{P}_{z,t}^{(r)}(f) \times I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)}{\sum_{t=1}^{T_r} I\left(u_{t,k}^{(r)} \in \text{non-speech}\right)}, \tag{5c}$$

where $\hat{P}_{z,t}^{(r)}(f)$ is the periodogram of $z_t^{(r)}$, which is defined as

$$\hat{P}_{z,t}^{(r)}(f) = \frac{1}{L}\left|z_t^{(r)}(f)\right|^2, \tag{6}$$

and L is the analysis length of the FFT operation; $I(\cdot)$ is the zero–one indicator function. Basing on Eq. (1b) of the assumed environment contamination model, the Wiener filter for the jth state of speech model and the rth training utterance is constructed and expressed by

$$W_{j,k}^{(r)}(f) = \frac{P_{y,j,k-1}(f)}{P_{y,j,k-1}(f) + P_{n,k}^{(r)}(f)}, \tag{7a}$$

where $P_{y,j,k-1}(f)$ is the average power density spectrum corresponding to the jth state of the bias-compensated speech HMM models. After forming all state-based Wiener filters, we calculate the enhanced signal by

$$y_{t,k}^{(r)}(f) = W_{u_t,k}^{(r)}(f) \times z_t^{(r)}(f), \quad \text{for } t = 1, \ldots, T_r \text{ and } u_t \neq \text{non-speech}. \tag{7b}$$

(2.2) SBR: Given with the segmentation information $(U_k^{(r)}, V_k^{(r)})$, estimate the signal bias and remove it from the intermediate signal $y_{t,k}^{(r)}(f)$ to obtain the environment-normalized speech estimate. The SBR method is realized by first transforming $y_{t,k}^{(r)}(f)$ to $y_{t,k}^{(r)}(m)$, then making a simplified assumption of $\Sigma_{z,j,q}^{(r)} =$ identity matrix in Eq. (A.11) of Appendix A to obtain

$$b_k^{(r)}(m) = \frac{\sum_{t=1}^{T_r} \left(y_{t,k}^{(r)}(m) - \mu_{x,u_{t,k}^{(r)},v_{t,k}^{(r)},k-1}(m) \right) \times I\left(u_{t,k}^{(r)} \notin \text{non-speech}\right)}{\sum_{t=1}^{T_r} I\left(u_{t,k}^{(r)} \notin \text{non-speech}\right)} \tag{8a}$$

and lastly removing the signal bias by

$$x_{t,k}^{(r)}(m) = y_{t,k}^{(r)}(m) - b_k^{(r)}(m). \tag{8b}$$

The third step of the REST algorithm is to re-estimate the speech HMM models $\Lambda_{x,k}$ and the average power density spectrum $\{P_{y,j,k-1}(f)\}_{j=1,\ldots,N_j}$ using, respectively, the enhanced speech signals $\{X_k^{(r)}(m)\}_{r=1,\ldots,R}$ and $\{Y_k^{(r)}(m)\}_{r=1,\ldots,R}$ based on the current segmentation information $\{(U_k^{(r)}, V_k^{(r)})\}_{r=1,\ldots,R}$, where N_j denotes the total number of states in HMM models.

The combination of all operations in above three steps can be interpreted as a sequential optimal estimation procedure listed in the following:

For iteration k

For utterance $r = 1$ to R, do

$$(U_k^{(r)}, V_k^{(r)}) = \arg\max_{(U^{(r)}, V^{(r)})} \Pr\left(Z^{(r)}, U^{(r)}, V^{(r)} \middle| \Lambda_{x,k-1}, \Lambda_{e,k-1}\right), \tag{9a}$$

$$\Lambda_{n,k}^{(r)} = \arg\max_{\Lambda_n^{(r)}} \Pr\left(Z^{(r)} \middle| \Lambda_n^{(r)}, \left(U_k^{(r)}, V_k^{(r)}\right)\right), \tag{9b}$$

$$Y_k^{(r)} = \arg\max_{Y^{(r)}} \Pr\left(Y^{(r)} \middle| Z^{(r)}, U_k^{(r)}, \Lambda_{n,k}^{(r)}, \{P_{y,j,k-1}\}_{j=1,\ldots,N_j}\right), \tag{9c}$$

$$b_k^{(r)} = \arg\max_{b^{(r)}} \Pr\left(Y_k^{(r)} \middle| b^{(r)}, \left(U_k^{(r)}, V_k^{(r)}\right), \Lambda_{x,k-1}\right), \tag{9d}$$

$$X_k^{(r)} = \arg\max_{X^{(r)}} \Pr\left(X^{(r)} \middle| Y_k^{(r)}, b_k^{(r)}\right). \tag{9e}$$

End loop for r

$$\{P_{y,j,k}\}_{j=1,\ldots,N_j} = \arg\max_{\{P_{y,j}\}_{j=1,\ldots,N_j}} \Pr\left(\{Y^{(r)}\}_{r=1,\ldots,R} \middle| \{P_{y,j}\}_{j=1,\ldots,N_j}, \left(U_k^{(r)}\right)_{r=1,\ldots,R}\right), \tag{9f}$$

$$\Lambda_{x,k} = \arg\max_{\Lambda_x} \Pr\left(\left\{X_k^{(r)}\right\}_{r=1,\ldots,R} \middle| \Lambda_x, \left(U_k^{(r)}, V_k^{(r)}\right)_{r=1,\ldots,R}\right). \tag{9g}$$

Repeat for k until the average likelihood score converges.

A similar idea was used in (Lim and Oppenheim, 1978; Hansen and Clements, 1991) to employ a sequential MAP estimation procedure in an iterative algorithm to sequentially estimate the linear prediction coefficients, gain, and the noise-free speech waveform for frame-level speech enhancement.

The REST algorithm can also be derived by using the EM algorithm (Dempster et al., 1977). So its convergence can be guaranteed. Detailed derivations of the EM procedure for estimating (Λ_x, Λ_e) is given in Appendix A.

Like other iterative algorithms, the REST algorithm must be initialized by giving an initial set of speech HMM models, an initial set of state averaged power density spectra, an initial channel bias vector, and an initial noise model. The initial speech HMM models and the initial state averaged power density spectra can be constructed by a conventional ML training algorithm using either an enhanced version of the given adverse-speech training set or another training set with high SNR. In the study, we adopt the former approach to use an enhanced speech training set obtained by subtracting the given initial noise model from the adverse-speech training set. The initial noise models are obtained from non-speech frames of the adverse-speech training set detected by an RNN-based speech segmentation method (Hong and Chen, 1997). It uses an RNN classifier, directly trained from adverse speech, to classify the input speech pattern into three broad-classes: *initial*, *final* and non-speech. The speech segmentation method has been shown to perform well in noisy environment (Hong et al., 1999). The initial bias vector is obtained by the SBR method using the above enhanced speech training set.

3. The PMC–SBC method for Mandarin base-syllable recognition

Mandarin Chinese is a tonal language. Each Chinese character is pronounced as a syllable with a tone. There are, in total, about 1300 syllables. If the tones are disregarded, there are only 411 phonologically allowed base-syllables. The phonetic structures of these 411 base-syllables are very regular and relatively simple as compared with English. A base-syllable can be decomposed into an optional *initial* and a *final*. There are in total 22 *initials* (including a null) and 39 *finals*. Although, the base-syllable set is only in medium size, its recognition is actually very difficult because it comprises many highly confusable sets. Specifically, all 411 base-syllables can be categorized into 39 confusable sets according to their *finals*. Like the English E-set, all base-syllables in each confusable set differ only in their *initial* consonants and are therefore difficult to be distinguished (Chang et al., 1993; Lee and Juang, 1996). Besides, cross-set confusion between these 39 sets are also easy to occur. Medial confusion and nasal-ending confusion are the two most commonly occurred types of cross-set confusion. Highly discriminative speech models are therefore needed to tackle the difficult task. In this study, a set of sub-syllable HMM models containing 100 3-state right-*final*-dependent *initial* models and 39 5-state context-independent *final* models is used as basic recognition units (Wang and Chen, 1998). In each state, a mixture Gaussian distribution with diagonal covariance matrices is used. The number of mixture in each state is variable and depends on the number of training samples, but a fixed maximum value is set for it. Besides, a single-state, single-mixture, utterance-dependent model is used for noise.

An integrated PMC-based Mandarin base-syllable recognition method, which is a modified version of the PMC method for additive and convolutional noise (Gales and Young, 1995; Nakamura et al., 1996) by additionally considering broad-class based likelihood compensation (Hong and Chen, 1997), is employed in this work to test the reference speech HMM models generated by the proposed REST training algorithm. It can be regarded as the combination of the PMC method and a signal bias compensation (SBC) method and is referred to as the PMC–SBC method. A block diagram of the new recognizer is displayed in Fig. 2. Each input testing utterance is first processed in the RNN-based Speech Segmentation (Hong and Chen, 1997) to detect non-speech frames. The RNN-based speech segmentation uses a three-layer simple RNN to discriminate each input frame among three broad-classes of *initial*, *final* and non-speech.

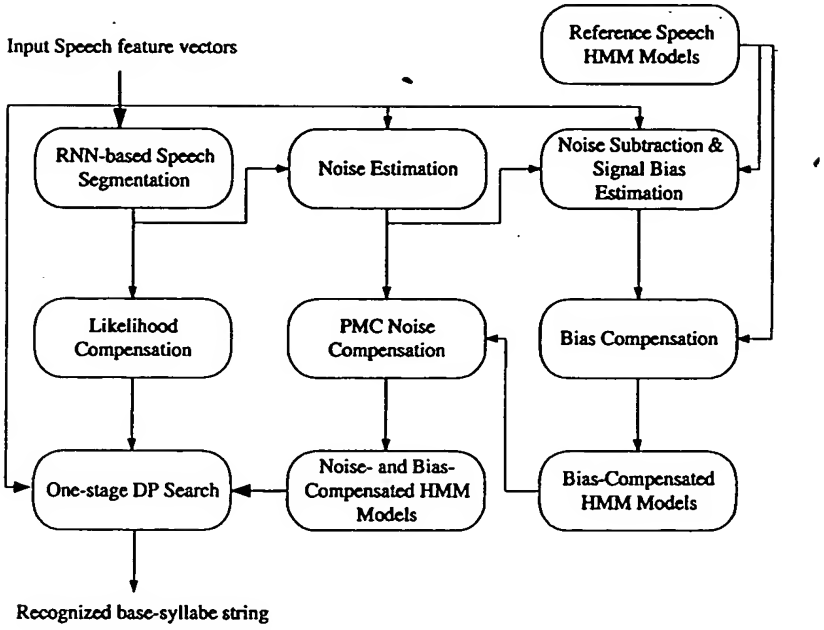


Fig. 2. A block diagram of the PMC–SBC method for testing the REST algorithm.

Non-speech frames are then detected by comparing the RNN non-speech output with a pre-determined threshold and used in the noise estimations to estimate the noise model. The input utterance is then processed in the Noise Subtraction and Signal Bias Estimation by first subtracting the noise model estimate to obtain an enhanced speech and then transforming to cepstrum domain to estimate the signal bias by the SBR method (Rahim and Juang, 1996). The SBR method estimates the signal bias by first encoding the feature vectors of the enhanced speech using a codebook and then calculating the average encoding residuals. The codebook is formed by collecting the mean vectors of mixture components of all reference speech HMM models. The bias estimate is then used in the Bias Compensation to convert all reference speech HMM models into bias-compensated speech HMM models. These models are then further converted, in the PMC Noise Compensation, into noise- and bias-compensated speech HMM models using the above noise model estimate. The PMC noise-compensation method used adopts the log-normal approximation (Gales and Young, 1993) for its noise-combination operator. These noise- and bias-compensated speech HMM models are then used in the One-stage DP Search to generate the recognized base-syllable sequence for the input adverse testing utterance. The One-stage DP Search uses a Viterbi search algorithm invoking with cumulative bounded-state-duration constraints (Wang and Chen, 1998) to accomplish its task with the help of the Likelihood Compensation. The likelihood compensation (LC) scheme used is the one proposed previously for improving the PMC-based recognition method for noisy Mandarin speech (Hong and Chen, 1997; Hong et al., 1999). The LC scheme uses the broad-class classification information, provided by the RNN outputs, to help reduce the recognition errors caused by the misalignments of syllable boundaries. Due to its importance, the LC scheme is briefly discussed as follows. Although the PMC method is effective on adapting the clean-speech HMM models to match with the testing noise environment, the discrimination capabilities of the noise-compensated HMM models are still subject to be degraded resulted from the noise perturbation on the distributions of the recognition features of speech patterns. This noise-perturbation effect will make all speech phones more difficult to be distinguished not

only to each other but also from the background noise. The PMC method can do nothing to compensate this effect. This noise-induced confusing effect was also confirmed in a recent study by Junqua et al. (1994) on a simple 10-digit noisy speech recognition task. They found that a large portion of recognition errors is owing to word boundary misalignments caused by the confusing between speech signals and the background noise. To partially cure the weakness of the PMC method, the LC scheme uses the broad-class classification information provided by the RNN to assist in the recognition. It directly takes the three RNN outputs as weighting factors to add additional scores to the log-likelihood scores of HMM states associated with the three broad classes, i.e.,

$$\rho_j^c(z_t) = \begin{cases} \rho_j(z_t) + \alpha \log(W_I(t)), & j \in \textit{initial}, \\ \rho_j(z_t) + \alpha \log(W_F(t)), & j \in \textit{final}, \\ \rho_j(z_t) + \alpha \log(W_N(t)), & j \in \text{non-speech}, \end{cases} \tag{10}$$

where $W_I(t)$, $W_F(t)$ and $W_N(t)$ are the *initial*, *final* and non-speech outputs of the RNN, $\rho_j(z_t)$ is the log-likelihood score of state j, and α is a scaling factor to control the degree of the likelihood compensation. It is noted that, if hard-decisions are performed in the broad-class classification to make $W_I(t)$, $W_F(t)$ and $W_N(t)$ become 0–1 functions, the LC scheme is equivalent to a restricted recognition search scheme in which only sub-syllables belonging to the detected broad-class are needed to be considered.

4. Evaluation

Performance of the proposed REST algorithm was evaluated on two multi-speaker Mandarin base-syllable recognition tasks. Due to the fact that the previous studies on robust training for eliminating the noise effect were still very few, we examined the effectiveness of the REST training algorithm on eliminating the noise effect in detail in the first task. Both the REST training algorithm and the PMC–SBC recognition method were simplified by discarding the parts related to the signal bias compensation. In the second task, the complete function of the REST algorithm on eliminating both the signal bias and noise effects was examined. In the following experiments, the base-syllable accuracy rate defined below was used to evaluate the recognition performance:

$$\text{base-syllable accuracy rate} = \left(1 - \frac{\text{Subs} + \text{Dels} + \text{Ins}}{\text{number of testing base-syllables}}\right) \times 100(\%), \tag{11}$$

where Subs, Dels and Ins denoted the numbers of substitution, deletion and insertion errors, respectively.

4.1. Performance evaluation I

In the first task, the performance of the REST algorithm on the adverse environment with only additive noise interference was examined. The noisy speech databases used in this study were generated by artificially adding noises to a clean-speech database composing of 1200 utterances of four speakers including two males and two females. Each utterance comprised several syllables and was pronounced in such a way that every syllable was clearly pronounced. The database contained in total 6197 syllables including 5124 training syllables and 1073 testing syllables. All speech signals were digitally recorded in a laboratory using a PC with a 16-bit Sound Blaster card and a head-set microphone. A sampling rate of 16 kHz was used. Two noisy-speech databases were artificially generated from the clean-speech database by adding noises of two different types including the Lynx helicopter noise from NOISEX-92 (Varga, 1993) and a computer-generated white Gaussian noise. For simplicity, these two noise types are referred to as Lynx and White noises, respectively. For each noise type, the training database contained three noisy-speech data sets of 12, 24 and 36 dB in SNR.

The open test used another three data sets for each noise type with 9, 18 and 30 dB in SNR. All speech signals were first pre-processed for each of 20 ms Hamming-windowed frame with 10 ms shift. Then, a set of 25 recognition features including 12 MFCC, 12 delta MFCC and a delta log-energy was computed for each frame. The maximum number of mixture components in each HMM state was set to be 5.

We first examined the efficiency of the speech HMM models generated by the REST algorithm using the F-ratio measure (Nicholson et al., 1997). The F-ratio is a measure of class separability in the acoustic feature space and can be roughly defined by

$$F\text{-ratio} = \frac{\text{variance of means}}{\text{mean of variances}}. \tag{12}$$

In this test, the classes were defined to include all states of the speech HMM models. The variance of means is the sample variance of all state means of these HMM models, and the mean of variances is the sample mean of all state variances. Obviously, a larger F-ratio measure indicates a larger separation among the states of the speech HMM models, which in turn roughly indicates that they have a higher discrimination capability. In the study, two schemes of the REST training algorithm with two different sets of initial models were tested. The first set of initial models, denoted as INIT1, was formed by the clean-speech HMM models, clean-speech state average power density spectra, and the exact noise models. Since INIT1 was an ideal model, the first scheme was not practical and hence was taken for reference only. The other set of initial models, denoted as INIT2, was a practical one and was generated by firstly segmenting all training utterances by the RNN-based speech segmentation method (Hong and Chen, 1997), secondly estimating the initial utterance-dependent noise models from non-speech frames of those training utterances, and lastly estimating the initial speech HMM models and the initial state average power density spectra from the enhanced version of the original training set obtained by subtracting the initial noise model. Figs. 3 and 4 show the feature-based F-ratio measures of the resulting HMM models for the two cases using Lynx and White noises, respectively. It can be seen from these two figures that the F-ratio measures for both schemes of the REST algorithm with INIT1 and INIT2 are comparable and are all better than the HMM_B models (to be defined later) trained by the conventional k-means algorithm. This is especially true for the lower-order recognition features. So the speech HMM models generated by the proposed REST algorithm are more compact and hence expected to possess better discrimination capability. Fig. 5 shows the learning curve of the REST algorithm. It can be found from Fig. 5 that the average log-likelihood score increases monotonically with respect to the iteration number. This empirically shows the convergence of the REST algorithm.

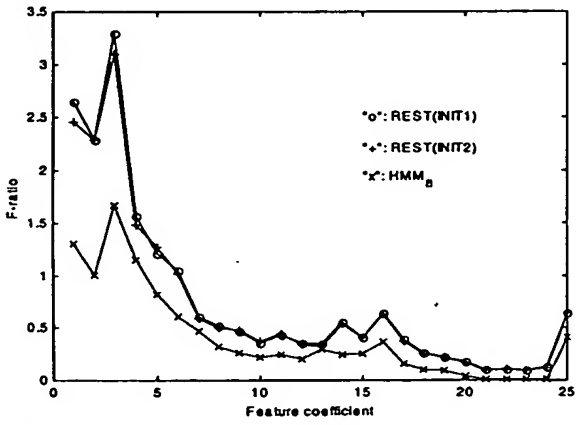


Fig. 3. The F-ratio measures of the speech HMM models trained from the noisy speech training database corrupted with Lynx noise.

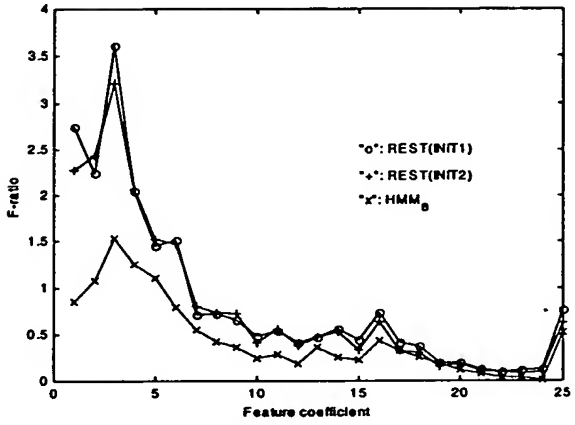


Fig. 4. The F-ratio measures of the speech HMM models trained from the noisy speech training database corrupted with White noise.

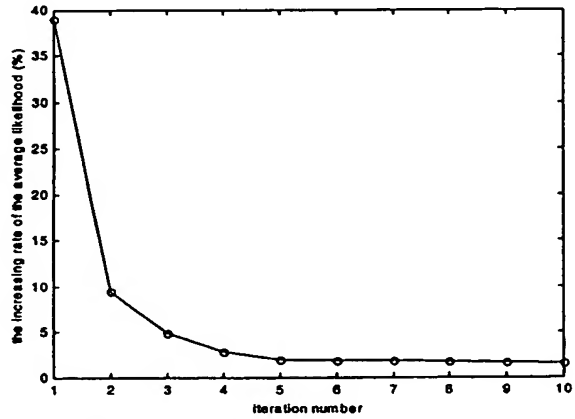


Fig. 5. The learning curve of the REST algorithm for the first task.

We then examined the recognition performance of the speech HMM models generated by the REST algorithm. The performance of the HMM method when both training and testing data were clean speech was also tested and taken as a benchmark. Its base-syllable recognition rate was 80.5%. In this test, four sets of reference speech HMM models were compared. They included:

M1. HMM_C: The HMM models trained from the clean-speech database by the ML-based segmental k-means algorithm.
M2. HMM_B: The HMM models trained from the noisy-speech database with three different SNRs by the ML-based segmental k-means algorithm.
M3. HMM_R: The HMM models trained from the noisy-speech database with three different SNRs by the proposed REST algorithm.
M4. HMM_M: The HMM models trained from a noisy-speech data set with SNR matched with the testing speech by the ML-based segmental k-means algorithm. That is, the HMM models trained from 9, 18 or 30 dB noisy-speech data set were used to recognize noisy speech with the same SNR.

For comparing the performances of these four sets of reference speech HMM models on noisy speech recognition, the following three recognition schemes were used:

S1-1. The 'NC' scheme: The conventional HMM recognition method without noise compensation.
S1-2. The 'PMC' scheme: The conventional PMC method (Gales and Young, 1993) with noise model being estimated based on RNN-based speech segmentation. Its noise-compensation operation used the log-normal approximation.
S1-3. The 'PMC/LC' scheme: An extended version of the 'PMC' scheme invoking with the likelihood compensation scheme. It is a degenerated version of the PMC–SBC method discussed in Section 3 with the parts related to signal-bias compensation being discarded.

Tables 1 and 2 show the experimental results of the open tests for the two cases using Lynx and White noises, respectively. It is noted that, in the implementation of the PMC recognition method using HMM_B as reference models, the mean of the estimated noise model, $\hat{\mu}_n^{(r)}(f)$, was intuitively modified by

$$\hat{\mu}_n^{(r)}(f) \leftarrow \begin{cases} \hat{\mu}_n^{(r)}(f) - \hat{\mu}_{n_0}(f), & \text{if } \hat{\mu}_n^{(r)}(f) > \hat{\mu}_{n_0}(f), \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

to count the noise effect embedded in the HMM_B models. Here $\hat{\mu}_{n_0}(f)$ is the noise mean of the training database estimated in the training process of generating the HMM_B models. From Tables 1 and 2, the following observations can be found:

O1. For HMM_B, the NC scheme performed fair for both noise types with SNR = 18 dB and SNR = 30 dB. But it performed very bad for both noise types with SNR = 9 dB.
O2. For HMM_M, the NC scheme performed very well for both noise types with all the three SNRs.
O3. For HMM_B, the PMC scheme performed only slightly better than the NC scheme for both noise types with SNR = 18 dB and SNR = 30 dB, and much better for SNR = 9 dB.
O4. The NC scheme with HMM_M performed better than the PMC scheme with HMM_C for both noise types with all the three SNRs.

Table 1
The recognition results of the open tests for noisy speech corrupted with Lynx noise (unit: %)

	HMM_B		HMM_C		HMM_R		HMM_M
SNR (dB)	NC	PMC	PMC	PMC/LC	PMC	PMC/LC	NC
9	−12.1	34.9	39.1	42.3	43.6	48.7	45.0
18	51.4	52.0	58.6	62.5	62.8	67.7	66.3
30	62.3	65.1	71.2	75.1	73.6	78.3	75.6

Table 2
The recognition results of the open tests for noisy speech corrupted with White noise (unit: %)

	HMM_B		HMM_C		HMM_R		HMM_M
SNR(dB)	NC	PMC	PMC	PMC/LC	PMC	PMC/LC	NC
9	−35.9	29.8	26.9	33.0	35.0	38.1	33.6
18	42.8	45.2	48.3	52.0	54.2	58.0	57.0
30	58.4	59.9	65.4	71.8	68.2	73.8	68.6

O5. For both PMC and PMC/LC schemes, HMM_R performed better than HMM_C for both noise types with all the three SNRs.
O6. For both HMM_R and HMM_C, the PMC/LC scheme performed much better than the PMC scheme.
O7. The PMC/LC scheme with HMM_R performed better than the NC scheme with HMM_M.

Based on these observations, the following conclusions can be drawn:

C1-1. From O1–O2, the conventional HMM method without noise compensation can be used in noisy speech recognition only when the noise level of the training data set is the same as that of the testing speech. If the training database contains noisy speech with diverse noise levels, its performance will degrade seriously.
C1-2. From O1–O3, the HMM models generated by the conventional k-means training algorithm are good for the NC scheme in the noise-level matched condition, fair in the noise-level interpolation condition, and bad in the noise-level extrapolation condition.
C1-3. From O1 and O3, the performance improvements for the HMM method using HMM_B reference models by the PMC noise compensation are very limited.
C1-4. From O4, the log-normal approximation of the noise-compensation operation used in the PMC scheme is not perfect.
C1-5. From O5 and C1-4, the REST algorithm is a very efficient training algorithm to generate noise-suppressed HMM models directly from a noisy speech database with diverse noise levels. The resulting HMM models perform very well in the PMC scheme for testing noisy speech with untrained noise levels. They are even better than the clean-speech HMM models for the PMC method when the noise-compensation operation is not perfect. So it is a very promising robust training algorithm.
C1-6. From O6–O7, the likelihood compensation scheme is very helpful for the PMC-based noisy speech recognition. Actually, the PMC/LC scheme using HMM_R reference models performed best in all cases of the test.

An extra test on noisy English digit recognition using the NOISEX-92 database (Varga and Steeneken, 1993) was performed to examine the validity of the proposed REST algorithm. The database contains utterances of isolated digits and digit triples uttered by one male and one female speakers. Here only the part of isolated-digit utterances was used. The database contains in total 400 digits including 200 training tokens and 200 testing tokens. Each testing utterance comprises 100 digits and was uttered in such a way that every digit was clearly pronounced. All speech signals were first pre-processed for each of 25 ms Hamming-windowed frame with 10 ms shift. Then, 12 MFCC were computed for each frame and taken as the recognition features. For each digit, an 8-state HMM model with observations in each state being modeled by a mixture Gaussian distribution was trained. The number of mixture components in each state was set to be 2. Besides, a single-state, single-mixture model was used for noise.

In the test, we considered the performance of the REST algorithm on the adverse environment with additive noise interference only. Noisy-speech databases were artificially generated from the clean-speech database by adding computer-generated white Gaussian noise. The noisy training database contained four data sets of 0, 6, 12 and 24 dB in SNR. The open test used another five data sets of −3, 0, 3, 9 and 18 dB in SNR. The same accuracy rate defined in Eq. (11) was used to evaluate the recognition performance. We note that the benchmark of the recognition performance achieved by the conventional ML-trained HMM method for the clean-speech case is 100%. Three recognition schemes used in the first test were compared. They included:

Table 3
The recognition results of the NOISEX-92 database corrupted by White noise (unit: %)

SNR (dB)	HMM_B–NC	HMM_C–PMC	HMM_R–PMC
−3	39.5	72.0	82.5
0	63.5	86.0	94.5
3	82.0	94.0	99.0
9	93.0	98.5	99.5
18	94.0	99.5	99.5

1. HMM_B–NC: The conventional HMM method without noise compensation using HMM models trained from noisy speech.
2. HMM_C–PMC: The PMC method using clean-speech HMM models.
3. HMM_R–PMC: The PMC method using HMM models trained by the REST algorithm.

Table 3 shows the experimental results. It can be seen from the table that HMM_B–NC performed the worst, HMM_C–PMC the next, and HMM_R–PMC the best. This result is consistent with what we have obtained in the first test of the study on adverse Mandarin speech recognition.

4.2. Performance evaluation II

In the second task, the performance of the REST algorithm on adverse environment with both channel bias and noise interferences was examined. A simulated telephone-speech database generated by corrupting a clean-speech database with both convolutional channel bias and additive white noise was used in this study. The clean-speech database was generated by 10 speakers including 8 males and 2 females. It was a super-set of the clean-speech database used in the first task with the same recording condition. It contained, in total, 3050 utterances including 2572 training utterances (12 800 syllables) and 478 testing utterances (2666 syllables). To generate the adverse-speech database, each clean-speech utterance was first corrupted by a computer-generated white Gaussian noise and then passed through a filter which simulated a telephone channel. This was realized simply by first adding the white noise in time domain and then adding the channel bias in frequency domain. It is noted that the assumed environment contamination model shown in Fig. 1 is still suitable for modeling the simulated database. In the training database generation, noises with levels of 12, 24 and 36 dB in SNR were separately added to three subsets of the clean-speech training database. These three subsets contained utterances of three, three and four speakers, respectively. In the testing database generation, noises with levels of 9, 18 and 30 dB in SNR were added to the whole clean-speech testing database. To simulate the channel variations on the telephone speech through the public switching network, a set of 227 simulated filters was generated from a large telephone-speech database provided by Chunghwa Telecommunication Laboratories. Each filter was obtained by performing a frame-based cepstrum average to the long utterance of a telephone call through the public switching network. Fig. 6 shows their frequency responses. Among these 227 channel filters, 195 were used to generate the training database while all others were used in the testing database generation. It is noted that the stationarity of the environment characteristics for each utterance is guaranteed in this simulated adverse-speech database via the use of utterance-dependent channel filter and noise level.

The same format of speech HMM models as the first task was used here. The only difference was that the maximal number of mixtures used in each HMM state was increased to 20. In the REST algorithm, the initial condition was generated from the same adverse training database by a four-step procedure. First, segment all training utterances by the RNN-based speech segmentation method (Hong and Chen, 1997). Second, estimate the initial utterance-dependent noise model from the non-speech frames of each training utterance. Third, estimate the initial speech HMM models and the initial state average power density

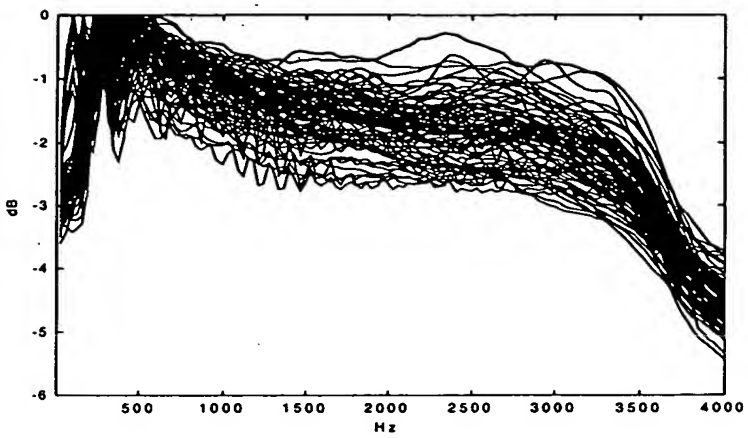


Fig. 6. The frequency responses of the simulated telephone channels.

spectra from the enhanced version of the original training set obtained by subtracting the initial noise models. Last, estimate the initial channel bias vectors from the same enhanced training set by the SBR method (Rahim and Juang, 1996).

In this test, the following recognition schemes were compared:

S2-1. The 'BASELINE' scheme: The conventional HMM method using the reference speech models trained directly from the adverse training database by the segmental k-means algorithm.
S2-2. The 'CLEAN' scheme: The PMC–SBC recognition method using the clean-speech reference HMM models, but without invoking the LC scheme.
S2-3. The 'REST-bias' scheme: The SBC recognition method using the reference HMM models trained by the REST algorithm without considering noise suppression.
S2-4. The 'REST-noise' scheme: The PMC recognition method using the reference HMM models trained by the REST algorithm without considering signal bias removal.
S2-5. The 'REST' scheme: The PMC–SBC recognition method using the REST-trained reference HMM models, but without invoking the LC scheme.
S2-6. The 'REST/LC' scheme: The PMC–SBC recognition method using the REST-trained reference HMM models.

Table 4 shows the base-syllable recognition results of these six schemes for adverse speech corrupted with channel bias and White noise. It can be found from Table 4 that, according to the recognition rate, these six schemes can be ordered as: REST/LC > REST > REST-noise or REST-bias > BASELINE > CLEAN. Based on the experimental results, the following conclusions can be made:

Table 4
The recognition results of the open tests for adverse speech corrupted with channel bias and White noise (unit: %)

SNR (dB)	BASELINE	CLEAN	REST-bias	REST-noise	REST	REST/LC
9	23.4	14.8	24.5	29.3	33.0	35.2
18	46.7	27.3	50.2	48.4	53.7	56.5
30	60.2	45.6	62.7	61.8	65.5	66.7

C2-1. The conventional HMM method using the reference models trained by the k-means algorithm performed fair in adverse speech recognition.
C2-2. The result that the CLEAN scheme performed much worse than the BASELINE scheme is mainly owing to the imperfection of the channel bias compensation performed in the SBC method. Actually, the CLEAN scheme was totally fail to compensate the mismatch between the testing speech and the clean-speech HMM model. This primarily resulted from the large deviation on the estimated signal bias from the real channel bias.
C2-3. Although the channel bias-compensation operation of the SBC method is imperfect, the REST training algorithm can still take its advantage by embedding it into the iterative training process to make the resulting HMM models more suitable to be used with the channel bias compensation of the testing process. This has been confirmed by the fact that both the REST-bias and REST scheme performed better than the BASELINE scheme.
C2-4. The HMM models generated by the REST algorithm which considers both noise suppression and signal bias removal are better than those obtained by the REST algorithm considering only noise suppression or signal bias.
C2-5. The likelihood compensation scheme is still effective on assisting in the adverse speech recognition.

A final test to check whether the REST training scheme is operable for clean-speech environment was lastly done. It is worthwhile to note that some robust training algorithms, designed for improving the performance of speech recognizers under adverse-speech environment, performed not well for clean-speech environment. In the test, two sets of HMM models were generated, respectively, by the conventional ML training method and by the REST training scheme using the same clean-speech database. The base-syllable recognition rate was 76.05% for the ML method and 76.24% for the REST scheme. This result confirmed that the REST algorithm did not degrade the system performance when the training data were clean speech.

5. Conclusions

A robust training algorithm for generating a set of speech HMM models directly from a training database collected in adverse environment for adverse speech recognition has been discussed in this paper. Its main advantage lies on the incorporation of the signal bias-compensation and PMC noise-compensation operations of a given robust adverse speech recognition method into its iterative training process so as to make the resulting speech HMM models more suitable to be used in the given robust adverse speech recognition method. Its effectiveness on generating robust speech HMM models has been confirmed by simulations. Experimental results showed that the HMM models it generated were even better than the clean-speech HMM models for use in the given robust adverse speech recognition method when the PMC noise-compensation and/or channel bias-compensation operations are imperfect. So it is a promising robust training algorithm.

Acknowledgements

This work was supported by the National Science Council of Taiwan under Contract no. NSC87-2213-E-009-056. The telephone-speech database was provided by the Chunghwa Telecommunication Laboratories.

Appendix A. The EM procedure for estimating $\{\Lambda_x, \Lambda_e\}$

Eq. (2) can be solved using an iterative EM procedure (Dempster et al., 1977) which tries to find the local optimal estimate of $\hat{\Theta} \equiv \{\hat{\Lambda}_x, \hat{\Lambda}_e\}$ with the following two intermediate parameter sequences involved: the hidden state sequences $\{U^{(r)} = (u_1^{(r)}, \ldots, u_{T_r}^{(r)})\}_{r=1,\ldots,R}$ and the mixture component sequences $\{V^{(r)} = (v_1^{(r)}, \ldots, v_{T_r}^{(r)})\}_{r=1,\ldots,R}$. The first (expectation) step of the EM procedure is to compute the auxiliary Q-function defined as

$$Q(\Theta, \hat{\Theta}_{k-1}) = E\left\{ \log L\left(\{Z^{(r)}, U^{(r)}, V^{(r)}\}_{r=1,\ldots,R} \middle| \Theta\right) \middle| \{Z^{(r)}\}_{r=1,\ldots,R}, \hat{\Theta}_{k-1} \right\}. \tag{A.1}$$

Here the subscript $k-1$ denotes the iteration index. In the second (maximization) step, new values of $\hat{\Theta}_k$ are computed based on the maximization of $Q(\Theta, \hat{\Theta}_{k-1})$:

$$\hat{\Theta}_k = \arg\max_{\Theta} Q(\Theta, \hat{\Theta}_{k-1}). \tag{A.2}$$

The detailed derivation of the EM procedure is described as follows.

Let

$$\Lambda_z^{(r)} = \begin{cases} G\left(\Lambda_x; \Lambda_n^{(r)}, b^{(r)}\right) & \text{for adverse-speech model,} \\ \Lambda_n^{(r)} & \text{for non-speech model} \end{cases} \tag{A.3}$$

be the environment-compensated HMM models, constructed from Λ_x and ($\Lambda_n^{(r)}$, $b^{(r)}$), for the rth observation utterance $Z^{(r)}$. Here $G(\cdot)$ denotes a mapping function that transforms Λ_x to match with the current environment of $Z^{(r)}$. By assuming that, in $\Lambda_z^{(r)}$, observations are mixture-Gaussian-distributed, we can calculate the mean vector $\mu_{z,j,q}^{(r)}$ and covariance matrix $\Sigma_{z,j,q}^{(r)}$ of the qth mixture component in the jth state of $\Lambda_z^{(r)}$, based on the assumed environment contamination model defined in Eqs. (1b) and (1c), by

$$\mu_{z,j,q}^{(r)} = \begin{cases} (\mu_{x,j,q} + b^{(r)}) \otimes \Lambda_n^{(r)}, & j \in \text{adverse-speech model,} \\ \mu_n^{(r)}, & j \in \text{non-speech model,} \end{cases} \tag{A.4a}$$

$$\Sigma_{z,j,q}^{(r)} = \begin{cases} \Sigma_{x,j,q} \otimes \Lambda_n^{(r)}, & j \in \text{adverse-speech model,} \\ \Sigma_n^{(r)}, & j \in \text{non-speech model,} \end{cases} \tag{A.4b}$$

where $\otimes$ denotes the PMC noise-compensation operator (Gales and Young, 1993), and $\mu_{x,j,q}$ and $\Sigma_{x,j,q}$ are, respectively, the mean vector and covariance matrix of the qth mixture component in the jth state of Λ_x. By further assuming that the state-based Wiener filtering is the inverse operation of the PMC (Gales and Young, 1993; Vaseghi and Milner, 1997), we can express the compensated cepstral mean $\mu_{z,j,q}^{(r)}$ in Eq. (A.4a) by (Gales and Young, 1993; Vaseghi and Milner, 1997)

$$\mu_{z,j,q}^{(r)} = \begin{cases} \mu_{x,j,q} + b^{(r)} + h_j, & j \in \text{adverse-speech model,} \\ \mu_n^{(r)}, & j \in \text{non-speech model,} \end{cases} \tag{A.5}$$

where h_j is the cepstral coefficients of the state-based Wiener filter of the jth state which is constructed from an estimate of the signal power density spectrum at the jth state and an estimate of the noise power density spectrum of the rth utterance.

Based on the above expression of $\Lambda_z^{(r)}$, the auxiliary Q-function can be rewritten as (Sankar and Lee, 1996)

$$\begin{aligned}
Q(\Theta, \hat{\Theta}_{k-1}) &= Q\left(\left(\Lambda_x, \{\Lambda_n^{(r)}, b^{(r)}\}_{r=1,\ldots,R}\right), \hat{\Theta}_{k-1}\right) \\
&= Q\left(\left(\{\Lambda_z^{(r)}\}_{r=1,\ldots,R}\right), \hat{\Theta}_{k-1}\right) \\
&= Q_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,k-1}^{(r)}(j,q)\log \Pr\left(z_t^{(r)}, u_t = j, v_t = q \middle| \Theta\right) \\
&= Q_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,k-1}^{(r)}(j,q)\log N(z_t^{(r)}; \mu_{z,j,q}^{(r)}, \Sigma_{z,j,q}^{(r)}),
\end{aligned} \tag{A.6}$$

where

$$\gamma_{t,k-1}^{(r)}(j,q) \equiv \Pr\left(z_t^{(r)}, u_t^{(r)} = j, v_j^{(r)} = q \middle| \hat{\Theta}_{k-1}\right) \tag{A.7}$$

is the probability of the observation $z_t^{(r)}$ produced from the qth mixture component of the jth state; N_j and N_q denote, respectively, the total numbers of states and mixture components; $N(\cdot)$ represents normal distribution; and Q_{k-1} is a function depending only on the transition probability and mixture probability of $\hat{\Lambda}_{z,k-1}^{(r)}$ (which are assumed to be the same as those of $\hat{\Lambda}_{x,k-1}$). But, due to the fact that it is generally difficult to derive a close form solution for the above joint maximization problem, a multi-stage sequential maximization procedure is employed to approximate the local optimum of $\hat{\Theta}_k$. In each stage, only one type of parameters is optimally estimated.

We first estimate the parameters of noise model $\hat{\Lambda}_{n,k}^{(r)}$ to maximize the Q-function in Eq. (A.6). They can be obtained by

$$\hat{\mu}_{n,k}^{(r)} = \frac{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,n,k-1}^{(r)}(j,q) z_t^{(r)}}{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,n,k-1}^{(r)}(j,q)}, \tag{A.8a}$$

$$\hat{\Sigma}_{n,k}^{(r)} = \frac{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,n,k-1}^{(r)}(j,q)\left(z_t^{(r)} - \hat{\mu}_{n,k}^{(r)}\right)\left(z_t^{(r)} - \hat{\mu}_{n,k}^{(r)}\right)^{\mathrm{T}}}{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,n,k-1}^{(r)}(j,q)}, \tag{A.8b}$$

where $\gamma_{t,n,k-1}^{(r)}(j,q) \equiv \gamma_{t,k-1}^{(r)}(j,q) I(j \in \text{non-speech})$ and $I(\cdot)$ is the zero–one indicator function.

We then estimate the signal bias $\hat{b}_k^{(r)}$. After replacing $\Lambda_n^{(r)}$ and Λ_x with $\hat{\Lambda}_{n,k}^{(r)}$ and $\hat{\Lambda}_{x,k-1}$, the Q-function becomes

$$\begin{aligned}
&Q\left(\left(\hat{\Lambda}_{x,k-1}, \left\{\hat{\Lambda}_{n,k}^{(r)}, b^{(r)}\right\}_{r=1,\ldots,R}\right), \hat{\Theta}_{k-1}\right) \\
&\quad = Q'_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\log N\left(z_t^{(r)}; \hat{\mu}_{x,j,q,k-1} + b^{(r)} - h_{j,k}, \Sigma_{z,j,q,k}^{(r)}\right) \\
&\quad = Q'_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\log N\left(\left(z_t^{(r)} + h_{j,k} - b^{(r)}\right); \hat{\mu}_{x,j,q,k-1}, \Sigma_{z,j,q,k}^{(r)}\right) \\
&\quad = Q'_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\log N\left(\left(y_{t,j,k}^{(r)} - b^{(r)}\right); \hat{\mu}_{x,j,q,k-1}, \Sigma_{z,j,q,k}^{(r)}\right),
\end{aligned} \tag{A.9}$$

where $\gamma_{t,s,k-1}^{(r)}(j,q) \equiv \gamma_{t,k-1}^{(r)}(j,q) I(j \in \text{speech})$; $h_{j,k}$ and $\Sigma_{z,j,q,k}^{(r)}$ are updated versions of h_j and $\Sigma_{z,j,q}^{(r)}$ with $\Lambda_n^{(r)}$ and Λ_x being replaced with $\hat{\Lambda}_{n,k}^{(r)}$ and $\hat{\Lambda}_{x,k-1}$, and $y_{t,j,k}^{(r)}$ is the Wiener-filtered version of $z_t^{(r)}$ at the jth state. By solving

$$\frac{\partial Q\left(\left(\hat{\Lambda}_{x,k-1}, \left\{\hat{\Lambda}_{n,k}^{(r)}, b^{(r)}\right\}_{r=1,\ldots,R}\right), \hat{\Theta}_{k-1}\right)}{\partial b^{(r)}} = 0, \tag{A.10}$$

the pth element of $\hat{b}_k^{(r)}$ can be obtained by (Sankar and Lee, 1996)

$$\hat{b}_k^{(r)}(p) = \frac{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\left(\Sigma_{z,j,q,k}^{(r)}(p,p)\right)^{-1}\left(y_{t,j,k}^{(r)}(p) - \hat{\mu}_{x,j,q,k-1}(p)\right)}{\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\left(\Sigma_{z,j,q,k}^{(r)}(p,p)\right)^{-1}}, \tag{A.11}$$

where $y_{t,j,k}^{(r)}(p)$ and $\hat{\mu}_{x,j,q,k-1}(p)$ denote, respectively, the pth elements of $y_{t,j,k}^{(r)}$, and $\hat{\mu}_{x,j,q,k-1}$, and $\Sigma_{z,j,q,k}^{(r)}(p,p)$ is the (p,p)th element of $\Sigma_{z,j,q,k}^{(r)}$.

We then estimate $\hat{\Lambda}_{x,k}$. After replacing $\Lambda_n^{(r)}$ and $b^{(r)}$ with $\hat{\Lambda}_{n,k}^{(r)}$ and $\hat{b}_k^{(r)}$, the Q-function becomes

$$Q\left(\left(\Lambda_x, \left\{\hat{\Lambda}_{n,k}^{(r)}, \hat{b}_k^{(r)}\right\}_{r=1,\ldots,R}\right), \hat{\Theta}_{k-1}\right) = Q_{k-1} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\log N\left(x_{t,j,k}^{(r)}; \mu_{x,j,q}, \Sigma_{x,j,q}\right), \tag{A.12}$$

where

$$x_{t,j,k}^{(r)} = y_{t,j,k}^{(r)} - \hat{b}_k^{(r)} = z_t^{(r)} + h_{j,k} - \hat{b}_k^{(r)} \tag{A.13}$$

is the signal bias-removed and Wiener-filtered signal of the jth state. The Q-function is now in the same form as that in the conventional EM algorithm for estimating HMM's parameters. So, the mean and covariance of $\hat{\Lambda}_{x,k}$ can be estimated in the same way by

$$\hat{\mu}_{x,j,q,k} = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)x_{t,j,k}^{(r)}}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)}, \tag{A.14a}$$

$$\hat{\Sigma}_{x,j,q,k} = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)\left(x_{t,j,k}^{(r)} - \hat{\mu}_{x,j,q,k}\right)\left(x_{t,j,k}^{(r)} - \hat{\mu}_{x,j,q,k}\right)^{\mathrm{T}}}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}\gamma_{t,s,k-1}^{(r)}(j,q)}. \tag{A.14b}$$

The HMM state transition probabilities and the mixture component coefficients can also be estimated by the standard EM method.

It can be verified that the Q-function will increase at each stage of the sequential maximization procedure, i.e.,

$$\begin{aligned}
Q\left(\hat{\Theta}_{k-1}; \hat{\Theta}_{k-1}\right) &= Q\left(\left(\hat{\Lambda}_{x,k-1}, \left\{\hat{\Lambda}_{n,k-1}^{(r)}, \hat{b}_{k-1}^{(r)}\right\}_{r=1,\ldots,R}\right); \hat{\Theta}_{k-1}\right) \\
&\leqslant Q\left(\left(\hat{\Lambda}_{x,k-1}, \left\{\hat{\Lambda}_{n,k}^{(r)}, \hat{b}_{k-1}^{(r)}\right\}_{r=1,\ldots,R}\right); \hat{\Theta}_{k-1}\right) \\
&\leqslant Q\left(\left(\hat{\Lambda}_{x,k-1}, \left\{\hat{\Lambda}_{n,k}^{(r)}, \hat{b}_{k}^{(r)}\right\}_{r=1,\ldots,R}\right); \hat{\Theta}_{k-1}\right) \\
&\leqslant Q\left(\left(\hat{\Lambda}_{x,k}, \left\{\hat{\Lambda}_{n,k}^{(r)}, \hat{b}_{k}^{(r)}\right\}_{r=1,\ldots,R}\right); \hat{\Theta}_{k-1}\right) \\
&= Q\left(\hat{\Theta}_k; \hat{\Theta}_{k-1}\right).
\end{aligned} \tag{A.15}$$

This in turn leads to an increase on the likelihood of the training data in each iteration (Dempster et al., 1977), i.e.,

$$L\left(\{Z^{(r)}\}_{r=1,\ldots,R}\middle|\hat{\Theta}_k\right) \geqslant L\left(\{Z^{(r)}\}_{r=1,\ldots,R}\middle|\hat{\Theta}_{k-1}\right). \tag{A16}$$

Hence, the EM procedure is guaranteed to converge.

In practical implementation, the above EM procedure needs to be modified by invoking with the segmental k-means algorithm (Juang and Rabiner, 1990) in order to increase its computational efficiency. It adds an additional pre-segmentation stage into the above iterative re-estimation procedure. In each iteration, all training utterances are first optimally segmented by the Viterbi algorithm (Forney, 1973) to determine the best state sequences $\{\hat{U}_k^{(r)}\}_{r=1,\ldots,R}$ and the best mixture component sequences $\{\hat{V}_k^{(r)}\}_{r=1,\ldots,R}$. Then, parameters of all models are re-estimated based on the given $\{\hat{U}_k^{(r)}, \hat{V}_k^{(r)}\}_{r=1,\ldots,R}$. All formulations of the above EM procedure listed in Eqs. (A.3)–(A.14) still hold except that $\gamma_{t,s,k-1}^{(r)}(j,q)$ and $\gamma_{t,n,k-1}^{(r)}(j,q)$ are now associated only with $\{\hat{U}_k^{(r)}, \hat{V}_k^{(r)}\}$ and hence all $\sum_{j=1}^{N_j}\sum_{q=1}^{N_q}$ in Eqs. (A.6), (A.8), (A.9), (A.11), (A.12) and (A.14) have to be taken away.

A final modification of the above re-estimation procedure is needed to replace the optimal signal bias estimation with the conventional SBR method. By making a simplified assumption of $\hat{\Sigma}_{z,j,q,k-1}^{(r)} = I$, the modified version of Eq. (A.11) can be reduced to Eq. (8a). This completes the derivations of the REST training algorithm.

References

Acero, A., Stern, R.M., 1990. Environmental robustness in automatic speech recognition. In: Proceedings of ICASSP-90, pp. 849–852.

Acero, A., Stern, R.M., 1991. Robust speech recognition by normalization of the acoustic space. In: Proceedings of ICASSP-91, pp. 893–896.

Anastasakos, T., McDonough, J., Makhoul, J., 1997. Speaker adaptive training: a maximum likelihood approach to speaker normalization. In: Proceedings of ICASSP-97, pp. 1043–1046.

Chang, P.-C., Chen, S.-H., Juang, B.-H., 1993. Discriminative analysis of distortion sequences in speech recognition. IEEE Trans. Speech and Audio Process. 1, 326–333.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc. 39, 1–38.

Ephraim, Y., 1992. Statistical-model-based speech enhancement systems. Proc. IEEE 80, 1526–1555.

Forney, G., 1973. The Viterbi algorithm. Proc. IEEE 61, 268–278.

Furui, S., 1992. Toward robust speech recognition under adverse conditions. In: Proceedings of the ESCA Workshop on Speech Processing in Adverse Conditions, pp. 31–24.

Gales, M.J.F., Woodland, P.C., 1996. Mean and variance adaptation within the MLLR framework. Comput. Speech and Language 10, 249–264.

Gales, M.J.F., Young, S.J., 1993. Cepstral parameter compensation for HMM recognition in noise. Speech Communication 12, 231–239.

Gales, M.J.F., Young, S.J., 1995. Robust speech recognition in additive and convolutional noise using parallel model combination. Comput. Speech and Language 9, 289–307.

Gales, M.J.F., Young, S.J., 1996. Robust continuous speech recognition using parallel model combination. IEEE Trans. Speech and Audio Process. 5, 352–359.

Gong, Y., 1995. Speech recognition in noisy environments: A survey. Speech Communication 16, 261–291.

Gong, Y., 1997. Source normalization training for HMM applied to noisy telephone speech recognition. In: Proceedings of EuroSpeech-97, Vol. 3, pp. 1555–1558.

Hansen, J.H.L., Clements, M.A., 1991. Constrained iterative speech enhancement with application to speech recognition. IEEE Trans. Signal Process. 39, 795–805.

Hermansky, H., Morgan, N., 1994. RASTA processing of speech. IEEE Trans. Speech and Audio Process. 2, 578–589.

Hong, W.-T., Chen, S.-H., 1997. A robust RNN-based pre-classification for Noisy Mandarin speech recognition. In: Proceedings of EuroSpeech-97, Vol. 3, pp. 1083–1086.

Hong, W.-T., Liao, Y.-F., Wang, Y.-R., Chen, S.-H., 1999. RNN-based speech segmentation and its applications to robust noisy Mandarin speech recognition. J. Acoust. Soc. Amer., revised.

Juang, B.-H., 1991. Speech recognition in adverse environment. Comput. Speech and Language 5, 275–294.

Juang, B.-H., Rabiner, L.R., 1990. The segmental K-means algorithm for estimating parameters of hidden Markov models. IEEE Trans. Acoust. Speech Signal Process. 38, 1639–1641.

Junqua, J.-C., Halton, J.-P., 1996. Robustness in Automatic Speech Recognition: Fundaments and Applications. Kluwer Academic Press, Boston, MA.

Junqua, J.S., Mak, B., Reaves, B., 1994. A robust algorithm for word boundary detection in the presence of noise. IEEE Trans. Speech and Audio Process. 2, 406–412.

Lee, C.-H., 1998. On stochastic feature and model compensation approaches to robust speech recognition. Speech Communication 25, 29–47.

Lee, C.-H., Juang, B.-H., 1996. A survey on automatic speech recognition with an illustrative example on continuous speech recognition of Mandarin. J. Comput. Linguist. Chinese Language Process. 1, 1–36.

Lim, J.S., Oppenheim, A.V., 1978. All-pole modeling of degraded speech. IEEE Trans. Acoust. Speech Sig. Process. 26, 197–210.

Liu, F.-H., Picheny, M., Srinivasa, P., Monkowaski, M., Chen, J., 1996. Speech recognition on Mandarin call home: a large vocabulary, conversational and telephone speech corpus. In: Proceedings of ICASSP-96, Vol. 1, pp. 157–160.

Lockwood, P., Boudy, J., 1992. Experiments with a Nonlinear Spectral Subtractor (NSS), Hidden Markov Models and the projection, for robust speech recognition in cars. Speech Communication 11, 215–228.

Mokbel, C.E., Chollet, G.F.A., 1995. Automatic word recognition in cars. IEEE Trans. Speech and Audio Process. 3, 346–356.

Minami, Y., Furui, S., 1996. Adaptation method based on HMM composition and EM algorithm. In: Proceedings of ICASSP-96, pp. 327–330.

Nakamura, S., Takigucgi, T., Shikano, K., 1996. Noise and room acoustics distorted speech recognition by HMM composition. In: Proceedings of ICASSP-96, Vol. 1, pp. 69–72.

Nicholson, S., Milner, B., Cox, S., 1997. Evaluating features set performance using the F-ratio and J-measures. In: Proceedings of EuroSpeech-97, Vol. 1, pp. 413–416.

Rahim, M., Juang, B.-H., 1996. Signal bias removal by maximum likelihood estimation for robust telephone speech recognition. IEEE Trans. Speech and Audio Process. 4, 19–30.

Sankar, A., Lee, C.-H., 1996. A maximum-likelihood approach to stochastic matching for robust speech recognition. IEEE Trans. Speech and Audio Process. 4, 190–202.

Varga, A., Steeneken, H.J.M., 1993. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. Speech Communication 12, 247–251.

Vaseghi, S.V., Milner, B.P., 1997. Noise compensation methods for hidden Markov model speech recognition in adverse environments. IEEE Trans. Speech and Audio Process. 5, 11–21.

Wang, Y.-R., Chen, S.-H., 1998. Mandarin telephone speech recognition for automatic telephone number directory service. In: Proceedings of ICASSP-98, Vol. 2, pp. 841–844.

Zhao, Y., 1996. Self-learning speaker and channel adaptation based on spectral variation source decomposition. Speech Communication 18, 65–77.

SEGMENTAL GPD TRAINING OF HMM BASED SPEECH RECOGNIZER

W. Chou, B.H. Juang and C.H. Lee

AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill,NJ 07974, U.S.A.

ABSTRACT

In this paper, we propose a new training algorithm, *segmental GPD training*, for hidden markov model (HMM) based speech recognizer using Viterbi decoding. This algorithm is based on the principle of minimum recognition error rate in which segmentation and discriminative training are jointly optimized. Various issues related to the special structure of HMMs in *segmental GPD training* are studied. We tested this algorithm on two speaker-independent recognition tasks. Our first experiment involves English E-set. *Segmental GPD training* was directly applied to HMMs generated from non-optimal uniform segmentation. A recognition rate of 88.7% was achieved on English E-set with whole word HMMs. Our second experiment involves connected digits TI-database. *Segmental GPD training* was applied to HMMs which were already trained using conventional training methods. A string recognition rate of 98.8% was achieved on 10-state whole word based HMMs through *segmental GPD training*.

1. INTRODUCTION

The use of hidden markov models (HMM's) with Viterbi decoding has become a prevalent approach in speech recognition, because of its simple algorithmic structure and its clear superiority over other alternative recognition schemes. But, in spite of its proved high performance for many recognition tasks, the conventionally trained HMMs are based mainly on the principle of statistical data fitting in terms of increasing the HMM likelihood. The optimality of this training criterion is conditioned on the availability of infinite amount of training data and the correct choice of the model. However, in reality, neither of these conditions are satisfied. The available training data is always limited, and the assumptions made by HMMs on speech production process are often inaccurate. As a consequence, the likelihood based training are not very effective for highly discriminative recognition applications and cannot guarantee optimal performance (i.e. minimum recognition error probability). This deficiency in the traditional training methods, namely, the lack of a direct relation with the recognition error rate motivates the recent effort of discriminative training [1–6].

Despite the algorithmic beauty of Viterbi decoding, its application to HMMs imposes several stringent constraints on training algorithms. A training algorithm for HMMs with Viterbi decoding must cope with the nature of the likelihood score on the optimal path, which has an intricate relation with HMM parameters.

Recently, discriminative training based on the "generalized probabilistic descent" (GPD) method has proved to be successful in many applications. In this paper, we propose a segmental based training method, *segmental GPD training*, for speech recognizer using hidden markov model and Viterbi decoding. The main features of our approach can be summarized as follows:

(1) The algorithm is based on the principle of minimum recognition error rate in which segmentation and discriminative training are jointly optimized.

(2) The algorithm can be initialized from a given HMM, regardless of whether it has been trained according to other criteria or directly generated from a training set with (non-optimal) uniform segmentation.

(3) The algorithm handles both errors and correct recognition cases in a theoretically consistent way, and is adaptively adjusted to achieve an optimal configuration with maximum possible separation between each confusing classes.

(4) The algorithm can be used either off-line or on-line with the ability of learning new features from any new training sources.

(5) The algorithm is consistent with HMM framework and does not require major modification of the current system. Moreover, it is theoretically justified to converge to a (at least locally) minimum point of the recognition error rate.

2. THE SYSTEM CONFIGURATION

In an HMM based recognizer. The continuous speech waveform is first blocked into frames and a discrete sequence of feature vectors, $X = \{x_0, x_1, \cdots, x_{T(x)}\}$, are extracted where T(x) corresponds to the total number of frames in the speech signal. We will identify the input speech utterance as its feature vector sequence $X = \{x_0, x_1, \cdots, x_{T(x)}\}$ without confusion.

In HMM, the observation probability density function of observing vector x in the j-th state of i-th word HMM is

0-7803-0532-9/92 $3.00

given by

$$b_j^i(x) = \sum_{k=1}^{L} c_{j,k}^i N(x, \mu_{i,j,k}, U_{i,j,k}); \quad (1)$$

which is a mixture of Gaussian distributions, where $c_{j,k}^i$ is the mixture weights and satisfies

$$\sum_{k=1}^{L} c_{j,k}^i = 1. \quad (2)$$

The optimal path under the Viterbi decoding is the one which attains the highest log-likelihood score. We denote Θ^i to be the optimal path of the input utterance X in i-th word HMM λ_i. Then, the log-likelihood score of the input utterance X along its optimal path in i-th model λ_i, $g_i(X, \lambda_i)$, can be written as

$$g_i(X, \Lambda) = \log f(X, \Theta^i \mid \lambda_i) \quad (3)$$

$$= \log b_{\theta_0^i}^i(x_0) + \sum_{t=1}^{T(X)} \log a_{\theta_{t-1}^i \theta_t^i}^i + \sum_{t=1}^{T(X)} \log b_{\theta_t^i}^i(x_{\theta_t^i}), \quad (4)$$

where θ_t^i is the corresponding state sequence along the optimal path Θ^i, x_t is the corresponding observation vector at time t, $T(X)$ is the number of frames in the input utterance X, $a_{\theta_{t-1}^i \theta_t^i}^i$ is the state transition probability from state θ_{t-1}^i to state θ_t^i.

The recognizer classifies the input utterance to i-th word W_i if and only if $i = \arg\max_j g_j(X, \lambda_j)$. If we define the classification error count function for i-th class as

$$\hat{l}_i(x, \lambda_i) = \begin{cases} 1 & X \in C_i \text{ and } i \neq \arg\max_j g_j(X; \lambda_j) \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

then the goal of training is to reduce the expected error rate

$$L(\Lambda) = E\left(\sum_{k=1}^{W} \hat{l}_k(X, \lambda_k)\right). \quad (6)$$

In practice, training result is often measured by the empirical error rate

$$L_0 = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{W} \hat{l}_k(X, \lambda_k). \quad (7)$$

However, direct minimization of the empirical error rate function has several serious deficiencies. It is numerically difficult, because classification error count function is not a continuous function. The inability of the empirical error rate function to distinguish near miss and barely correct cases may impair the performance of the recognizer on the independent test data set. Viterbi decoding also adds one more complexity here, because under the Viterbi decoding, the form and the value of the empirical error rate function varies with segmentation determined by the HMM parameters. A set of numerically optimal HMM parameters based on the current segmentation does not maintain its optimality under a different segmentation, unless a good convergence result can be proved.

3. SEGMENTAL GPD TRAINING OF HMMS

Our approach to this problem is to embed both the classification error count function and the decision rule into a smooth functional form–loss function. For each class, we introduce a misclassification measure, which provides a distance information concerning the correct class and all other competing classes. These misclassification measures are included in the loss function. In *segmental GPD training*, the loss function is constructed through the following steps:

(1) Let $g_j(X, \lambda_j)$ be the log-likelihood score of the input utterance in the j-th word model. Define the misclassification measure for each class i by

$$d_i(X, \Lambda) = -g_i(X, \lambda_i) + \log\left[\frac{1}{W-1} \sum_{j \neq i} e^{g_j(X, \lambda_j)\eta}\right]^{\frac{1}{\eta}} \quad (8)$$

where η is a positive number, W is the total number of classes. If $\mu_i(j)$ represents a discrete measure defined on the finite integer set $\{j \mid j \neq i \text{ and } 1 \leq j \leq W\}$ with equal mass $\frac{1}{W-1}$ on each integer j, then

$$\left[\frac{1}{W-1} \sum_{j \neq i} e^{g_j(X, \lambda_j)\eta}\right]^{\frac{1}{\eta}}$$

$$= \left(\int e^{g_j(X, \lambda_j)\eta} d\mu_i(j)\right)^{1/\eta} = \|e^{g_j(X, \lambda_j)}\|_\eta \quad (9)$$

is an L^η norm approximation to $\max_{j \neq i} e^{g_j(X, \lambda_j)} = \|e^{g_j(X, \lambda_j)}\|_\infty$ as $\eta \to \infty$. Misclassification measure $d_i(x, \Lambda) > 0$ indicates a misclassification has been observed, which means that $g_i(X, \lambda_i)$ is significantly smaller than $\max_{j \neq i} g_j(X, \lambda_j)$. Moreover, the sign and absolute value of the misclassification measure $d_i(x, \Lambda)$ implies the near miss and barely correct cases.

(2) Define the smoothed loss function for each class by

$$l_i(d_i(X, \Lambda)) = \frac{1}{1 + e^{-\gamma d_i(X, \Lambda)}}. \quad (10)$$

(3) Define the loss function for entire training population by

$$l(X, \Lambda) = \sum_{k=1}^{W} l_k(X, \Lambda) 1(X \in W_k). \quad (11)$$

By controlling parameters η and γ, we can have an accurate smoothed approximation to the classification error count function. Therefore, minimization of the expected loss of this specially designed loss function is directly linked to the minimization of the error probability. Note that the misclassification measure of (8) takes into account all the competing classes. This makes an efficient multi-class adjustment training algorithm possible. Generalized probabilistic decent (GPD) algorithm adjusts the model parameters Λ recursively according to

$$\Lambda_{n+1} = \Lambda_n - \epsilon_n U_n \nabla l(X_n, \Lambda_n), \quad (12)$$

where U_n is a properly designed positive definite matrix, $\{\epsilon_n : n \geq 1\}$ is a sequence of positive numbers, and $\nabla l(X_n, \Lambda_n)$ is the gradient vector of the loss function $l(X, \Lambda)$ at the n-th training sample X_n. This algorithm is proved to converge, provided that $\sum_{n=1}^{\infty} \epsilon_n = \infty$ and $\sum_{n=1}^{\infty} \epsilon_n^2 < \infty$ (see [8] for a detailed discussion). Our emphasize on the general form of GPD algorithm given in (12) has its intention. In *segmental GPD training*, a properly designed positive definite matrix sequence U_n in (12) is not only instrumental but crucial as will be discussed in next section.

4. PARAMETER TRANSFORMATIONS

In *segmental GPD training*, the HMM parameters are adaptively adjusted according to (12). A diagram of this training procedure is illustrated in Fig. 1. However, due to the special structure of HMMs, the parameters of HMMs must satisfy certain constraints. These constraints are not guaranteed to be satisfied by GPD algorithm. Thus, some treatments are necessary.

Instead of using a complicated constrained GPD algorithm, we apply *segmental GPD training* on transformed HMM parameters. These transformations have the purpose of maintaining all constraints on the HMM parameters during the process of *segmental GPD training*. The following transformations are used in our approach:

(1) Logarithm of the variance

$$\tilde{\sigma}^2_{i,j,k,d} = \log \sigma^2_{i,j,k,d} \tag{13}$$

where $\sigma^2_{i,j,k,d}$ is the variance of the i-th word, j-th state, k-th mixture and d-th dimension.

(2) Transformed logarithm of the mixture wrights

$$\tilde{c}_{i,j,k} = \left(\frac{e^{c_{i,j,k}}}{\sum_{l=1}^{L} e^{c_{i,j,l}}}\right) \tag{14}$$

where L is the total number of the mixture weights in the j-th state in the i-th word model.

(3) Transformed logarithm of the transition probability

$$\tilde{a}_{i,j} = \frac{e^{a_{i,j}}}{\sum_{k=1}^{M} e^{a_{i,k}}} \tag{15}$$

where M is total number of states in i-th word model.

A critical step of *segmental GPD training* lies in how to handle the problem of small variance. Variances in HMMs can differ by as many as 10^4 to 10^6 times. Using a constant step size ϵ_n for all HMM parameters will not produce the desired result, because the sensitivity of the mean parameter adjustment is determined by the size of the variance. The same ϵ_n can be too large for some mean parameters and too small for others. For a moderate complex HMM recognition system, there are 10^4 to 10^5 parameters to be adjusted simultaneously at each iteration. Without a theoretical guide, it is almost impossible to observe the desired performance improvement within finite steps.

In order to compensate for this vast difference in sensitivity, a carefully designed positive definite matrix U_n is crucial. The positive definite matrix U_n used in our approach is a diagonal matrix

$$diag(\sigma_1^2(n), \cdots, \sigma_D^2(n)).$$

for each state, where $\sigma^2(n)$ is the variance of HMM at time n. This corresponds to a GPD training on the normalized mean parameter $\frac{m_d}{\sigma_d}$ which takes care the sensitivity issue [8].

5. EXPERIMENTAL EVALUATION

The proposed *segmental GPD algorithm* is evaluated on two speaker independent tasks. In both experiments, whole word based HMMs were used. Each HMM is a left-to-right HMM with Gaussian mixture state observation densities. The covariance matrix in each HMM is a diagonal matrix. The feature vectors used in the experiments consist of 38-elements, with 12 cepstrum coefficients, 12 delta-cepstrum coefficients, 12 delta-delta spectrum coefficients, the delta log energy and delta-delta log energy [9].

Our first experiment involves the English E-set (b,c,d,e,g,p,t,v,z). The speech signal was recorded from 100 native Americans including 50 male and 50 female through local dialed-up telephone lines. Every talker spoke each word twice to produce two data sets. One was used for training and the other for testing. We started with untrained 10-state, 5-mixture, left-to-right, whole word based HMMs, directly generated from the non-optimal uniform segmentation. The recognizer has a recognization rate of 76% on the testing set (89% on the training set). In 10 iterations of the *segmental GPD training*, the recognizer achieved a recognition rate of 88.3% on the testing set (99.6% on the training set). We also tested *segmental GPD training* on 15-state, 3-mixture, left-to-right whole word based HMMs generated from uniform segmentation. The recognition rate of the recognizer before *segmental GPD training* is 73.3% on the testing set (86.3% on the training set). The recognizer achieved a recognition rate of 88.7% on the testing set (100% on the training set). In both cases, a 50% error rate reduction was achieved by *segmental GPD training*. These results are the best results reported so far on this data set using the whole word based HMMs. Fig. 2 illustrates the performance improvement during the training process of 15-state, 3-mixture HMMs.

Our second experiment is related to the speaker independent, TI-database of connected digit utterances. The digits string of TI-database has a random length from 1 to 7. The speech signal was recorded from various region of the United States. It contains 8565 strings for training and 8578 strings for testing. The model we used was a 10-state, 64-mixture, whole word based HMM model which was trained by the conventional method and was shown to achieve top performances. Our *segmental GPD training* was based on the word level error, in which the word boundary information was obtained by running a path on all training utterances. Then, *segmental GPD training* was applied on

Recognizer	Original	GPD training
Data Set	TI-digit	TI-digit
Substitution Error	60	53
Total String Error	113	104
Recognition Rate	98.7%	98.8%

Table 1: Recognition result of TI-data base

the segmented word utterances. After three iterations, a string error rate reduction of 8% on this well trained HMM model was obtained. The string recognition rate on the testing data set is 98.8%. As illustrated in Table 1, the improvement was mainly due to the reduction of the substitution errors, which is exactly the effect of the word level error based training.

6. SUMMARY AND DISCUSSION

In this paper, we have proposed a new training method, *segmental GPD training*, for HMM based recognizer using Viterbi decoding. We investigated its performance on two speaker independent recognition tasks using HMMs directly generated from non-optimal uniform segmentation and HMMs trained by conventional methods. *Segmental GPD training* is based on the principle of minimum recognition error rate with a theoretically justified convergence property. In our approach, both classification error count and the decision rule are embedded into a smooth functional form, and segmentation and discriminative training are jointly optimized for the goal of minimum recognition error rate. Various issues related to the special structure of HMMs in *segmental GPD training* are studied. We demonstrated the effectiveness of the proposed training algorithm in isolated word and connected digit recognition applications. Further research and experiments on sub-word based system are in progress.

Acknowledgements

The authors would like to thank Dr. L. Rabiner and Dr. B. Atal for their suggestions and supports during the process of this research.

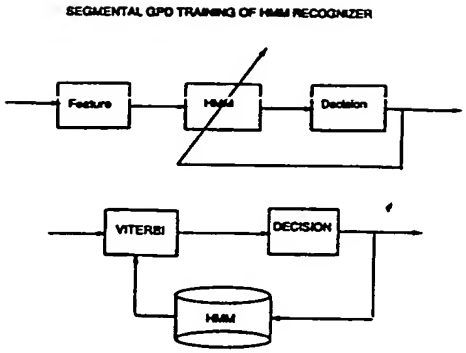


Figure 1: Diagram of *segmental GPD training*.

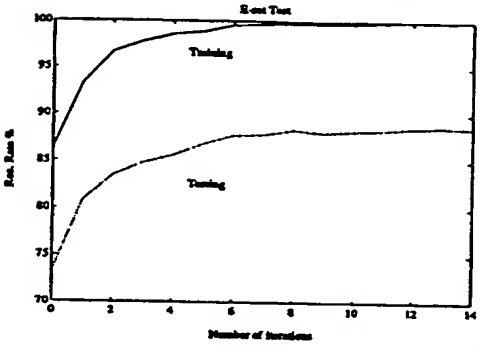


Figure 2: Recognition curve of *segmental GPD training*(88.7% on testing data set).

REFERENCES

[1] L.R. Bahl, P.F. Brown, P.V. De Souza and R.L. Mercer, "A new algorithm for the estimation of hidden Markov model parameters", *Proc. ICASSP88*.

[2] K-F Lee and S. Mahajan "Corrective and reinforcement learning for speaker-independent continuous speech recognition", *Computer and Language*, 1990, pp231-245.

[3] S. Katagiri and C.H. Lee, "A new HMM/LVQ Hybrid Algorithm for Speech Recognition", *Proc. Globecom90*, November 1990.

[4] S. Katagiri, C.H. Lee and B.H. Juang, "New discriminative algorithms based on the generalized probabilistic descent method", *Proc. IEEE-SP Workshop on Neural Network for Signal Processing*, Princeton, Sept. 1991.

[5] P.C. Chang, S.H. Chen and B.H. Juang, "Discriminative Analysis of Distortion Sequences in Speech Recognition", *Proc. ICASSP91*, pp. 549–552.

[6] W. Chou and B.H. Juang, "Adaptive discriminative learning in pattern recognition", Technical Report of AT&T Bell Laboratory

[7] A Ljolie, Y. Ephraim and L.R. Rabiner, "Estimation of hidden markov model parameters by minimizing empirical error rate", *Proc. ICASSP89*, pp. 709–712.

[8] W. Chou, B.H Juang and C.H. Lee, "Segmental GPD training of hidden markov model with minimum recognition error rate criterion", in preparation.

[9] J. Wilpon, C.H. Lee and L.R. Rabiner, "Improvements in connected digits recognition using higher order spectral and energy features", *Proc. ICASSP91*, pp. 349–352.

[10] L.R. Rabinar, "A tutorial on Hidden Markov Models and selected applications in speech recognition", *Proc. IEEE, vol. 77, No.2, pp 257–285, 1989.*

[11] L. R. Rabiner, B. H. Juang, S. E. Levinson, M. M. Sondhi, "Recognition of Isolated Digits Using Hidden Markov Models with Continuous Mixture Densities," *AT&T Technical Journal*, 64(6), pp. 1211-1233, July-Aug. 1985.

Speech Communication 12 (1993) 231–239
North-Holland

Cepstral parameter compensation for HMM recognition in noise

M.J.F. Gales and S.J. Young

Cambridge University Engineering Department, Trumpington Street, Cambridge, UK

Received 19 February 1993

Abstract. This paper describes a method of adapting a continuous density HMM recogniser trained on clean cepstral speech data to make it robust to noise. The technique is based on parallel model combination (PMC) in which the parameters of corresponding pairs of speech and noise states are combined to yield a set of compensated parameters. It improves on earlier cepstral mean compensation methods in that it also adapts the variances and as a result can deal with much lower SNRs. The PMC method is evaluated on the NOISEX-92 noise database and shown to work well down to 0 dB SNR and below for both stationary and non-stationary noises. Furthermore, for relatively constant noise conditions, there is no additional computational cost at run-time.

Zusammenfassung. Dieser Artikel beschreibt eine Methode zur Anpassung eines auf versteckten Markov Modulen basierenden Erkennungssystems mit kontinuierlicher Dichte (aufgenommen über Parameter, die die normale Sprache darstellen), um das System bei Vorhandensein von Lärm sicherer zu machen. Diese Methode, die auf der Kombination von parallelen, Modellen beruht, ermöglicht die Kombination von gepaarten Lärm- und Sprachzuständen, um daraus eine Reihe von kompensierten Parametern zu bilden. Dies ist eine Verbesserung, im Vergleich zu den Kompensationsmethoden des Mittelwertes, da diese Methode auch die Anpassung der Standardabweichungen ermöglicht, wodurch wesentlich geringere Rauschabstände berücksichtigt werden können. Diese Methode wird basierend auf der Datenbank NOISEX-92 bewertet. Wir zeigen, daß diese Methode bei einem Rauschabstand von 0 dB oder kleiner im Rahmen von stationärem und nicht stationärem Lärm gute Ergebnisse liefert. Außerdem gibt es bei dieser Methode bei relative konstanten Lärmbedingungen keine zusätzliche Rechenzeit in der Testphase.

Résumé. Cet article décrit une méthode dont le but est d'adapter un système de reconnaissance basé sur des HMM à densité continue (appris sur des paramètres cepstraux représentant de la parole normale) pour rendre le système plus robuste en présence de bruit. Cette méthode, fondée sur la combinaison de modèles parallèles, permet de combiner les états appariés de bruit et de parole pour fournier un ensemble de paramètres compensés. Ceci est une amélioration par rapport à des méthodes de compensation de la moyenne cepstrale car cette méthode permet aussi d'adapter les variances, ce qui permit de prendre en compte des rapports signal sur bruit beaucoup plus faibles. Cette méthode est évaluée sur la base de données NOISEX-92. Nous montrons qu'elle donne de bons résultats pour un rapport signal sur bruit de 0 dB ou inférieur dans le cadre de bruits stationnaires et non-stationnaires. De plus, pour des conditions de bruit relativement constante cette méthode n'ajoute aucun temps de calcul en phase de test.

Keywords. Speech recognition; noise compensation; AMN; PMC.

1. Introduction

As speech recognition technology moves from the laboratory to real applications, there is a growing need to make systems which are robust to a wide range of background noises. Many different methods have been studied for achieving noise robustness (Juang, 1991; Furui, 1992), most of which can be classified into one of two major approaches.

Firstly, the corrupted speech input signal can be preprocessed prior to the pattern matching stage in an attempt to enhance the signal-to-noise ratio (SNR). The methods used in this approach

include spectral subtraction (Boll, 1979; Van Compernolle, 1989; Lockwood and Boudy, 1992) and spectral mapping (Sorensen, 1991; Cung and Normandin, 1992). The main difficulty with this approach is that it must rely solely on exploiting knowledge about the interfering noise since there can be no a priori knowledge of what will be said.

The second class of methods attempt to modify the pattern matching stage itself in order to account for the effects of noise. Methods in this approach include noise masking (Klatt, 1979, Holmes and Sedgewick, 1986; Mellor and Varga, 1992), the use of robust distance measures (Mansour and Juang, 1988; Carlson and Clements, 1991), state-based filtering (Beattie and Young, 1991), cepstral mean compensation (Chen, 1987; Berstein and Shallom, 1991; Beattie and Young, 1992) and HMM decomposition (Moore, 1986; Varga and Ponting, 1989; Kadirkamanathan, 1992).

This paper is concerned with the latter approach to noise robustness. In particular, a scheme based on parallel model combination (PMC) will be described (Gales and Young, 1992). PMC is based on the assumption that knowledge of both the noise and the speech should be exploited to gain maximal effect. This implies that noise compensation should take place in the pattern matching stage where knowledge of the speech to be recognised is embedded in the stored patterns. In the case of an HMM recogniser, this implies that the compensation must be state-based to allow stationarity of the speech component to be assumed.

The PMC approach is closely related to the HMM decomposition approach referenced above. There are, however, two important differences. Firstly, HMM decomposition operates in the log filter-bank domain rather than in the preferred cepstral domain. It therefore lacks the advantages of the cepstral transform in terms of parameter decorrelation and compactness. Furthermore, it requires the state variances to be diagonal, compounding the problem of correlation between the filter-bank channels. Secondly, it carries a high computational cost since the output probabilities have to be calculated from both the noise and speech distributions at run-time. PMC, on the other hand, works directly in the cepstral domain and, depending on the variability of the noise sources, the additional run-time overhead can be as low as zero.

The remainder of this paper is organised as follows. In the next section, the basic theory of PMC is outlined and its relationship to an existing method of cepstral mean compensation is discussed. Section 3 then considers the practical issues of covariance approximation and multiple state noise modelling. Section 4 describes an evaluation of the PMC method on the NOISEX noise database (Varga et al., 1992) and finally, Section 5 presents some conclusions.

2. Parallel model combination

2.1. Basic theory

PMC assumes that the speech to be recognised is modelled by a set of continuous density HMMs which have been trained using clean speech data. Similarly, the interfering background noise is also modelled by a continuous density HMM which will initially be assumed to consist of a single state. All signals are represented by Mel-Frequency Cepstral Coefficients (MFCCs).

Given a clean speech HMM with state output distributions characterised by means and variances $\{\mu_i, \Sigma_i\}$, the noise mean and covariance $\{\bar{\mu}, \bar{\Sigma}\}$ is combined with each state i in turn to calculate a set of compensated distribution parameters $\{\hat{\mu}_i, \hat{\Sigma}_i\}$. The basis of the calculation is the assumption that the speech and noise are additive in the linear power domain. Noisy speech can therefore be regarded as being generated by the clean speech HMM operating in parallel with the noise HMM. These models can be combined to give a compensated noisy speech HMM by mapping the distribution parameters back into the linear spectral domain, finding the parameters of the sum of the two distributions and then mapping back to the MFCC domain. This Parallel Model Combination process is summarised in Figure 1.

Notice that since there is only one noise state there is no ambiguity as to which noise state should be combined with each speech model state. Modelling non-stationary noise will, however, re-

quire multiple state noise HMMs and this is discussed further in the next section.

The details of the mapping procedure are as follows. Let μ^c and Σ^c be the mean and covariance of any state output distribution in the cepstral domain. Cepstral parameters are derived from the log spectrum by the discrete cosine transform which can be represented by a matrix C. Since this transform is linear, the corresponding distribution μ^l and Σ^l in the log spectral domain are given straightforwardly by

$$\mu^l = C^{-1}\mu^c, \tag{1}$$

$$\Sigma^l = C^{-1}\Sigma^c(C^{-1})^T. \tag{2}$$

If the distributions in the cepstral and log spectral domains are assumed to be Gaussian, then the distributions in the linear domain will be log normal. The i-th component of the mean μ in the linear domain is then given by

$$\mu_i = E[e^{x_i}]$$

$$= \int_{\mathscr{R}^n} \frac{1}{(\sqrt{2\pi})^n |\Sigma^l|^{1/2}} \times \exp\left(x_i - \tfrac{1}{2}(x-\mu^l)^T(\Sigma^l)^{-1}(x-\mu^l)\right) dx, \tag{3}$$

where $\mathscr{R}^n$ is the region of all possible acoustic observation vectors x in the log spectral domain. As shown in Appendix A this may be simplified to give

$$\mu_i = e^{\mu_i^l + \Sigma_{ii}^l/2}. \tag{4}$$

Similarly, the variance in the linear domain can be calculated from the expectation

$$E[e^{x_i}\, e^{x_j}] = \int_{\mathscr{R}^n} \frac{1}{(\sqrt{2\pi})^n |\Sigma^l|^{1/2}} \times \exp\left(x_i + x_j - \tfrac{1}{2}(x-\mu^l)^T(\Sigma^l)^{-1} \times (x-\mu^l)\right) dx. \tag{5}$$

In Appendix A this is shown to reduce to

$$E[e^{x_i}\, e^{x_j}] = \mu_i \mu_j\, e^{\Sigma_{ij}^l}. \tag{6}$$

Hence,

$$\Sigma_{ij} = E[e^{x_i}\, e^{x_j}] - E[e^{x_i}]E[e^{x_j}]$$

$$= \mu_i \mu_j \left[e^{\Sigma_{ij}^l} - 1\right]. \tag{7}$$

The above mapping is used to derive the distribution parameters in the linear spectral domain for each pair of speech and noise states. From the assumption that the speech and noise are independent and additive, the combined mean and covariance are given by

$$\hat{\mu} = g\mu + \bar{\mu}, \tag{8}$$

$$\hat{\Sigma} = g^2\Sigma + \bar{\Sigma}, \tag{9}$$

where (μ, Σ) are the speech model parameters and $(\bar{\mu}, \bar{\Sigma})$ are the noise model parameters. The factor g is a gain matching term introduced to account for the fact that the level of the origin clean speech training data may be different from that of the noisy speech.

If the combined distribution in the linear spectral domain is assumed to be approximately log

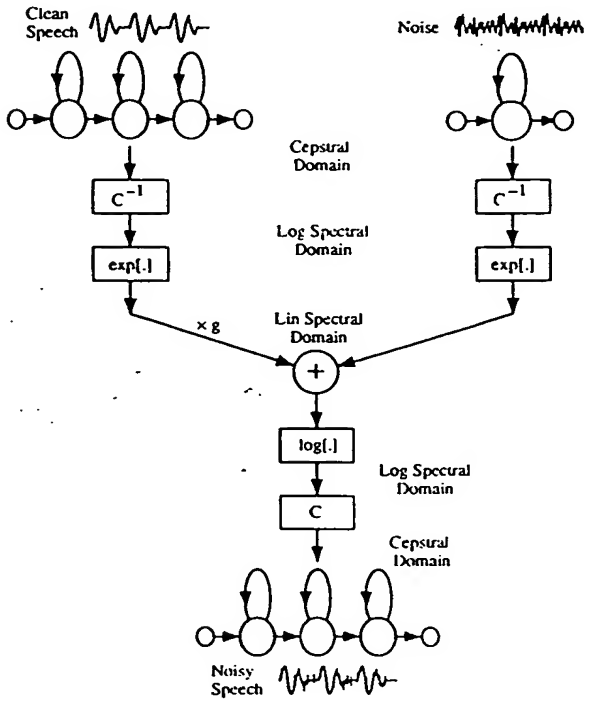


Fig. 1. Parallel model combination.

normal, the above process can be straightforwardly inverted. Firstly, the linear domain parameters are mapped back to the log domain by

$$\hat{\mu}_i^l = \log(\hat{\mu}_i) - \tfrac{1}{2}\log\left[\frac{\hat{\Sigma}_{ii}}{\hat{\mu}_i^2} + 1\right], \tag{10}$$

$$\hat{\Sigma}_{ij}^l = \log\left[\frac{\hat{\Sigma}_{ij}}{\hat{\mu}_i\hat{\mu}_j} + 1\right], \tag{11}$$

and secondly, back to the cepstral domain by

$$\hat{\mu}^c = C\hat{\mu}^l, \tag{12}$$

$$\hat{\Sigma}^c = C\hat{\Sigma}^l C^T. \tag{13}$$

To prepare an HMM recogniser for operation in a particular stationary noise environment, a one state noise model is trained on samples of the background, and the average noisy speech signal energy E_{ns} and the average background noise energy E_n are estimated. Using a gain matching term given by

$$g = \frac{E_{ns} - E_n}{E_s}, \tag{14}$$

where E_s is the average energy of the clean training speech, the noise state mean $\bar{\mu}$ and covariance $\bar{\Sigma}$ are used to calculate compensated output distribution parameters $\{\hat{\mu}_i, \hat{\Sigma}_i\}$ for every state i of every model. In a practical system, this compensation process would be repeated during idle periods so that slowly changing noise or signal levels could be tracked.

2.2. Relationship to Wiener filtering

The PMC method just described is related to existing state-based Wiener Filtering cepstral mean compensation methods (Berstein and Shallom, 1991; Beattie and Young, 1992). The basic assumption of the WF method is that the speech associated with a given HMM state is stationary with power spectrum $S(f)$. Hence, given knowledge of the noise power spectrum $N(f)$, a matched filter can be designed by

$$W(f) = \frac{gS(f)}{gS(f) + N(f)}, \tag{15}$$

where g is a gain matching term as described above. This equation may be rewritten in terms of the noisy speech observations $\hat{S}(f)$ as

$$S(f) = W(f)\hat{S}(f), \tag{16}$$

which in turn leads to the equivalent relation in terms of cepstral means,

$$\mu = w + \hat{\mu}, \tag{17}$$

where μ is the cepstral mean of the clean estimate and therefore corresponds to the mean obtained when training on clean speech and $\hat{\mu}$ is the desired compensated cepstral mean which is matched to the noisy speech. The cepstral transform w of $W(f)$ can therefore be regarded as an estimate of the mean shift needed to transform the clean speech mean into a noisy speech mean.

The estimate of the noisy speech mean given by eq. (17) can be written equivalently in the log power domain as

$$\hat{\mu}^l = \mu^l - w^l. \tag{18}$$

From eq. (15), assuming without loss of generality that $g = 1$ and replacing $S(f)$ by the clean mean μ and $N(f)$ by the noise mean $\bar{\mu}$ gives in the log power domain

$$w^l = \log(\mu) - \log(\mu + \bar{\mu}). \tag{19}$$

Under the assumptions used in the PMC method, substituting eqs. (19) and (10) into eq. (18) gives

$$\hat{\mu}_i^l = \log(\mu_i + \bar{\mu}_i) - \tfrac{1}{2}\log\left(\frac{\Sigma_{ii}}{\mu_i^2} + 1\right), \tag{20}$$

whereas the PMC method gives

$$\hat{\mu}_i^l = \log(\mu_i + \bar{\mu}_i) - \tfrac{1}{2}\log\left(\frac{\Sigma_{ii} + \bar{\Sigma}_{ii}}{(\mu_i + \bar{\mu}_i)^2} + 1\right). \tag{21}$$

From this it can be seen that for the WF method the speech variance is taken account of implicitly, because the HMM is trained in the log domain. However, the noise is estimated in the linear domain and its variance is assumed to be zero. Thus, the PMC method should have advantages for noise with a significant variance. Furthermore, the PMC also yields compensated covari-

ances and should therefore be more effective at very low SNRs.

3. Practical implementation

3.1. Covariance approximation

The procedure described in the previous section assumes that full covariance matrices are used. In practice, it is common to assume that observation vector components are independent so that diagonal covariance matrices can be used. This reduces the amount of training data required and reduces the run-time computational requirements. The PMC compensation scheme, however, yields full covariance matrices even when the initial clean speech and noise distributions are both diagonal. Furthermore, the independence assumption for some noise sources is not justified and hence it is beneficial in some cases to use a full covariance matrix for the noise model.

In order to avoid the run-time penalty of using full covariance matrices in the compensated models, one of two simple approximations can be used. Firstly, the full covariance matrices can be made diagonal by simply setting the off-diagonal terms to zero. As shown below, this has little effect on performance.

Secondly, a so-called *fixed variance* can be used (Paul, 1987) whereby the diagonal variance of the entire clean speech data is used for all state variances (and never re-estimated). However, when used for continuous speech, the fixed variance should be scaled to match the determinant of the noise covariance so that the normalising constants in the inter-word noise model probability distribution and the within-word speech model probability distributions are equalised. This is not, of course, necessary when the fixed variance is used with the noise model as well. However, this simplification is unnecessary since there is usually sufficient data to reliably estimate the noise variance. Fixed variance works well for moderate SNRs and has the advantage that it can be implemented as a global scaling of the input data, thereby significantly reducing the run-time cost.

3.2. Non-stationary noise

In order to deal with non-stationary noise, a multi-state noise model can be used. In this case, the same PMC method applies but now it is no longer possible to know a priori which noise state to combine with each speech model state. Hence, all combinations must be computed and the optimal sequence decoded at run-time. The standard method of dealing with this is to use a 3-dimensional Viterbi Decoding scheme based on the recursion (Moore, 1986).

$$\Phi_t(j, v) = \max_{i,u} \Phi_{t-1}(i, u) a_{ij} \tilde{a}_{uv} b_{jv}(x_t), \qquad (22)$$

where $\Phi_t(j, v)$ is the maximum joint probability of being in state j of the speech model and state v of the noise model at time t, and observing t sequence x_1 to x_t. The combined output probability $b_{jv}(\cdot)$ in the PMC case corresponds to the distribution obtained by combining state j of the speech model with state v of the noise model. Note that if the clean speech HMMs contain a total of M states and the noise model has N states, then the compensated recogniser will have $M \times N$ states. Fortunately, 2 or 3 states are usually sufficient for the noise model.

When the noise model is ergodic (i.e. fully connected), then the above 3-D decoding scheme can be synthesised using a standard 2-D decoder operating on an expanded model whereby each original speech state i is replaced by N compensated states $i1, i2, \ldots, iN$ with transition probabilities given by

$$\hat{a}_{iu,jv} = a_{ij}\tilde{a}_{uv}. \qquad (2$$

Figure 2 illustrates this for the case of a 3 state word model combined with a 2 state noise model. The obvious advantage of this scheme is that it enables standard HMM recognisers to work in non-stationary background noise. Note, however, that there is a small loss of information at the word boundaries since the effective self-loop transition probabilities for the noise states cannot be preserved exactly. In practice, this seems to be of no real significance.

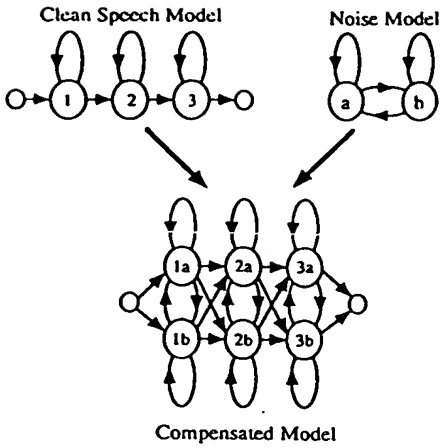


Fig. 2. Constructing a compensated model for non-stationary background noise.

4. Evaluation on NOISEX-92

In this section, a number of experiments using the NOISEX-92 Database are reported (Varga et al., 1992). This data was pre-processed using a 25 msec Hamming window and a 10 msec frame period. For each frame, a set of 15 MFCC coefficients were computed. The zeroth cepstral coefficient is computed and stored since it is needed in the PMC mapping procedure. However, it is subsequently dropped in the actual recognition process.

NOISEX contains one male and one female speaker uttering both isolated digits and digit triples. The test data for each speaker and condition consists of a single file containing all of the test tokens spoken in sequence with a *silence* interval between each. Here only the male isolated digits were used of which there are 100 training tokens and 100 test tokens. Five of the eight possible background noises were used. Three stationary noises: F16 fighter, Lynx helicopter and a car; and 2 non-stationary noises: machine gun and operations room. For each case, the noise is mixed with the clean speech at 5 levels in the range +18 dB to −6 dB.

For each digit, a single mixture continuous density HMM with 8 emitting states was trained using the clean speech data only. The topology for all models was left–right with no skips and diagonal covariance matrices were assumed throughout. For each test condition, a single state noise HMM was trained using the silence intervals of the test files with, except where stated, a full covariance matrix. Recognition used a standard connected-word Viterbi decoder constrained by a syntax consisting of silence followed by a digit in a loop. Thus, no explicit end-point detector was used and insertion/deletion errors occurred as well as classifications errors. The results are in terms of % accuracy, where for N tokens, S substitution errors, D deletion errors and I insertion errors accuracy is calculated as $[(N-S-D-I)/N]\times 100\%$. The error counts themselves were calculated by using a DP string matching algorithm between the recognised digit sequence and the reference transcription. Since the NOISEX data is synthetic, the gain matching term g can be set exactly. Hence, for all the experiments reported here $g=1$ was used. Note, however, that in practice, PMC is not sensitive to the exact choice of g. All of the training and testing used version 1.4 of the portable HTK HMM Toolkit (Young, 1992), with suitable extensions to perform the PMC.

Figure 3 shows the performance of the PMC compensated HMMs compared to the standard uncompensated HMMs. In this case, the compensated models use full covariance matrices. As can be seen, the compensation is effective down to at least 0 dB SNR. Table 1 shows that the effect of dropping the off-diagonal terms from the compensated covariance matrices to restore the runtime recogniser to using diagonal variances is

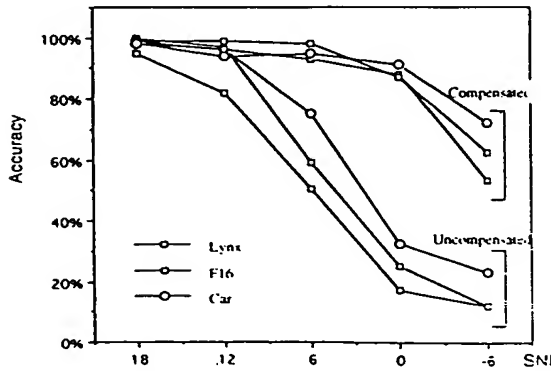


Fig. 3. PMC versus baseline HMM recogniser for male isolated digits in Lynx, F16 and Car noise.

Table 1
Comparison of full covariance matrices (Full) with diagonal approximation (Diag) and scaled fixed variance (Fixed)

SNR dB	Lynx			F16			Car		
	Full	Diag	Fixed	Full	Diag	Fixed	Full	Diag	Fixed
−6	53	46	30	62	57	38	72	70	49
0	88	90	68	87	83	87	91	94	85
+6	93	97	97	98	96	98	95	96	96
+12	96	100	100	99	100	100	94	96	100
+18	98	99	100	99	100	100	98	99	100

minimal. It also shows that good performance can also be obtained from the scaled fixed variance method although this is not quite as good at very low SNRs. Note that without the scaling, the performance at 0 dB and below worsens considerably. For example, the accuracy for the F16 noise at 0 dB drops from 87% to 51% when the scaling is removed, and at −6 dB it drops from 38% to 21%.

A related issue to covariance approximation is the question of whether diagonal covariances are adequate for the noise models. Table 2 compares recognition performance for the three stationary noise sources for full covariance and diagonal covariance noise models. In both cases, a diagonal covariance was used with the compensated models. The results show that for these noise sources at least, full covariance models are unnecessary.

Figure 4 shows the performance of PMC with non-stationary background noise. For these noises, more than 1 noise state is essential. For the operations room noise, there is a very small further improvement using a 4 state model compared to a 2 state model, but for machine gun noise the performance is unchanged for 2 or more noise states.

Table 2
Comparison of full covariance noise model (FullN) with diagonal covariance noise model (DiagN)

SNR dB	Lynx		F16		Car	
	FullN	DiagN	FullN	DiagN	FullN	DiagN
−6	46	48	57	54	70	72
0	90	92	83	82	94	94
+6	97	98	96	96	96	96
+12	100	100	100	100	96	96
+18	99	99	100	100	99	99

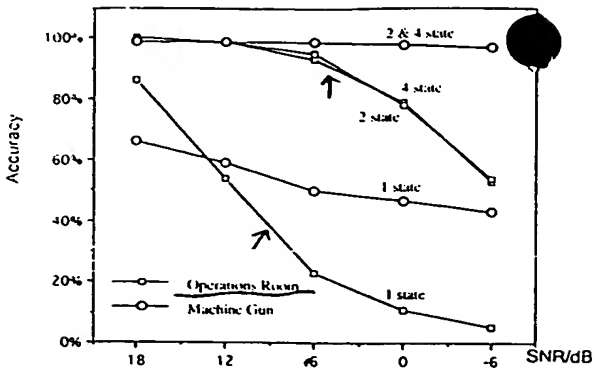


Fig. 4. Effect of number of noise states for modelling non-stationary noise.

5. Conclusions

This paper has discussed the use of Parallel Model Combination (PMC) parameter compensation for transforming a set of HMM word models trained on clean data into a set of models which can be used under a specific set of noise conditions.

The PMC approach has been evaluated on the synthetic NOISEX-92 database and shown to give a significant improvement to the noise robustness of an HMM-based recogniser. Two practical aspects of PMC have been discussed and evaluated. Firstly, simple diagonalisation of the compensated full covariance matrices by zeroing off-diagonal terms shows no significant loss in performance. Also, where a single global variance must be used, then a scaled fixed variance scheme gives adequate performance. Secondly, for non-stationary noise, an ergodic noise model with 2 or more noise states is both necessary and effective.

The performance of an HMM recogniser for very high SNRs is not affected by PMC compensation. The use of delta (difference) coefficients would further improve the results reported here for SNRs of +6 dB and better. However, the use of uncompensated delta coefficients at low SNRs would seriously damage performance. An effective compensation scheme for delta coefficients is

therefore also needed and work is in progress in this area.

The overall conclusion is that PMC is a very simple yet effective approach to dealing with noise in an HMM based system. For relatively static noise conditions, a once-only adjustment of the HMM means and variances is all that is required, and hence the run-time cost is zero. In practice, it might be expected that a new noise model and consequent set of parameter adjustments would be recomputed periodically when the recogniser was idle. For more rapidly changing noise, multiple state noise models must be used. These require the original speech model states to be duplicated to include all combinations of noise and speech state, but in practice the duplication factor can be kept very small.

Acknowledgments

Part of the work reported here was performed in the Esprit II ARS Project 2101. Thanks are also due to Andrew Varga at DRA, Malvern for supplying the NOISEX-92 CD ROM.

Appendix A. Derivation of log-normal expectations

The expectation of e^{x_i} is defined in Section 2.1 as

$$\mathrm{E}[e^{x_i}] = \int_{\mathscr{R}^n} K \exp\Big(x_i - \tfrac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1} \times (x-\mu)\Big)\,\mathrm{d}x$$

$$= K\int_{\mathscr{R}^n} \exp\Big(x_i - \tfrac{1}{2}x^{\mathrm{T}}\Sigma^{-1}x + x^{\mathrm{T}}\Sigma^{-1}\mu - \tfrac{1}{2}\mu^{\mathrm{T}}\Sigma^{-1}\mu\Big)\,\mathrm{d}x, \quad (24)$$

where K is the usual normalising constant and $\mu = \mathrm{E}[x]$. This integral may be evaluated as follows. Let

$$y = \mu + \Sigma e^i, \quad (25)$$

where e^i is a unit vector with the i-th component unity and all other components zero, i.e. it has the property

$$x_i = x^{\mathrm{T}} e^i. \quad (26)$$

From the definition of y,

$$\exp\Big(-\tfrac{1}{2}(x-y)^{\mathrm{T}}\Sigma^{-1}(x-y)\Big) = \exp\Big(-\tfrac{1}{2}x^{\mathrm{T}}\Sigma^{-1}x + x^{\mathrm{T}}\Sigma^{-1}y - \tfrac{1}{2}y^{\mathrm{T}}\Sigma^{-1}y\Big)$$

$$= \exp\Big(x_i - \tfrac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu) - \mu - \tfrac{1}{2}\Sigma_{ii}\Big) \quad (27)$$

Hence,

$$\mathrm{E}[e^{x_i}] = K\int_{\mathscr{R}^n} \exp\Big(-\tfrac{1}{2}(x-y)^{\mathrm{T}}\Sigma^{-1}(x-y) + \mu_i + \tfrac{1}{2}\Sigma_{ii}\Big)\,\mathrm{d}x$$

$$= e^{\mu_i + \Sigma_{ii}/2}\left\{K\int_{\mathscr{R}^n} \exp\Big(-\tfrac{1}{2}(x-y)^{\mathrm{T}}\Sigma \times (x-y)\Big)\,\mathrm{d}x\right\}. \quad (28)$$

The integrand of the term in braces is a simple Gaussian and the integral is therefore unity, hence the required result follows:

$$\mathrm{E}[e^{x_i}] = e^{\mu_i + \Sigma_{ii}/2}. \quad (29)$$

The expectation of $e^{x_i}e^{x_j}$ follows in an identical way, but this time the substitution

$$y = \mu + \Sigma(e^i + e^j) \quad (30)$$

is used leading to the result

$$\mathrm{E}[e^{x_i}\,e^{x_j}] = \mu_i\mu_j\,e^{\Sigma_{ij}}. \quad (31)$$

References

V.L. Beattie and S.J. Young (1991). "Robust speech recognition using hidden Markov models and state-based filtering", *Internat. Conf. Acoust. Speech Signal Process.*, Toronto, May 1991.

V. Beattie and S.J. Young (1992). "Hidden Markov model state-based cepstral noise compensation", *Proc. ICSLP*, Banff, Canada, pp. 519–522.

A. Berstein and I. Shallom (1991). "An hypothesised Wiener filtering approach to noisy speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Toronto, S14.9

S. Boll (1979), "Suppression of acoustic noise in speech using spectral subtraction", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. 27, pp. 113–120.

B.A. Carlson and M.A. Clements (1991), "Application of a weighted projection measure for robust HMM-based speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Toronto, S14.11, pp. 921–924.

Y. Chen (1987), "Cepstral domain stress compensation for robust speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, 17.7.1, pp. 717–720.

H.M. Cung and Y. Normandin (1992), "Noise adaptation algorithms for robust speech recognition, *Proc ESCA Workshop on Speech Processing in Adverse Conditions*, Cannes-Mandelieu, November 1992, pp. 171–174; also in *Speech Communication*, Vol. 12, No. 3, July 1993, pp. 267–276.

S. Furui (1992), "Toward robust speech recognition under adverse conditions", *Proc. ESCA Workshop on Speech Processing in Adverse Conditions*, Cannes-Mandelieu, November 1992, pp. 31–42.

M.J.F. Gales and S.J. Young (1992), "An improved approach to the hidden Markov model decomposition of speech and noise", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, San Francisco, March 1992, S35.1, pp. 233–236.

J.N. Homes and N. Sedgewick (1986), "Noise compensation for speech recognition using probabilistic models", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Tokyo, pp. 741–744.

B.H. Juang (1991), "Speech recognition in adverse environments", *Comput. Speech Language*, Vol. 5, No. 3, pp. 275–294.

Kadirkamanathan (1992), "Hidden Markov model decomposition recognition of speech in noise: A comprehensive experimental study", *Proc. ESCA Workshop on Speech Processing in Adverse Conditions*, Cannes-Mandelieu, November 1992, pp. 187–190.

D.H. Klatt (1979), "A digital filter bank for spectral matching", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, pp. 573–576.

P. Lockwood and J. Boudy (1992), "Experiments with a Non-linear Spectral Subtractor (NSS), Hidden Markov models and the projection, for robust speech recognition in cars, *Speech Communication*, Vol. 11, Nos. 2–3 (Eurospeech '91), pp. 215–228.

D. Mansour and B.H. Juang (1988), "A family of distortion measures based upon projection operation for robust speech recognition", *Proc Internat. Conf. Acoust. Speech Signal Process.*, New York, S1.5, pp. 36–39.

B.A. Mellor and A.P. Varga (1992), "Noise masking in the MFCC domain for the recognition of speech in background noise", *Proc. Inst. Acoustics Autumn Conf. on Speech and Hearing*, Vol. 14, Part 6, pp. 361–368.

R.K. Moore (1986), Signal decomposition using Markov modelling techniques, RSRE Memo No 3931, Royal Signals and Radar Establishment, Malvern, July 1986.

D.B. Paul (1987), "A speaker-stress resistant HMM isolated word recogniser", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Dallas, 17.6.1, pp. 713–716.

H.B.D. Sorensen (1991), "A cepstral noise reduction multilayer neural network", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Toronto, S14.14, pp. 933–936.

D. Van Compernolle (1989), "Spectral estimation using a log-distance error criterion applied to speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Glasgow, 21.S6.2, pp. 258–261.

A. Varga and K. Ponting (1989), "Control experiments on noise compensation in hidden Markov model based continuous word recognition", *Proc. Eurospeech*, Paris, September 1989, pp. 167–170.

A. Varga, H.J.M. Steeneken, M. Tomlinson and D. Jones (1992), The NOISEX-92 study on the effect of additive noise on automatic speech recognition, Technical Report, DRA Speech Research Unit, Malvern, England.

S.J. Young (1992), HTK Version 1.4: Reference Manual and User Manual, Cambridge University Engineering Department, Speech Group, August 1992.